



45565

PATENT

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Sung-Jin Park et al.

Serial No.: 10/634,746

Filed: August 6, 2003

For: TURBO DECODING APPARATUS

:  
:  
:  
:  
:  
:  
:  
:

Group Art Unit:

TRANSMITTAL OF PRIORITY DOCUMENT

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

In order to perfect the claim for priority under 35 U.S.C. §119(a), the Applicants herewith submit a certified copy of Korean Patent Application No. 2002-46410, as filed on August 6, 2002. Should anything further be required, the Office is asked to contact the undersigned attorney at the local telephone number listed below.

Respectfully submitted,

Peter L. Kendall  
Attorney of Record  
Reg. No.: 46,246

Roylance, Abrams, Berdo & Goodman, L.L.P.  
1300 19<sup>th</sup> Street, N.W., Suite 600  
Washington, D.C. 20036-2680  
(202) 659-9076

Dated: January 16, 2004



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto is a true copy from the records of the Korean Intellectual Property Office.

출원 번호 : 10-2002-0046410  
Application Number

출원 년 월 일 : 2002년 08월 06일  
Date of Application  
AUG 06, 2002

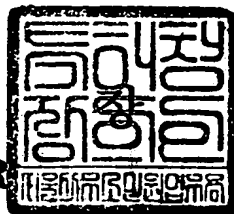
출원인 : 삼성전자주식회사  
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



2003      08      06      일  
          년      월

특      허      청

COMMISSIONER



## 【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0003
【제출일자】	2002.08.06
【국제특허분류】	H04L
【발명의 명칭】	터보 복호화 장치
【발명의 영문명칭】	TURBO DECODING APPARATUS
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이건주
【대리인코드】	9-1998-000339-8
【포괄위임등록번호】	1999-006038-0
【발명자】	
【성명의 국문표기】	박성진
【성명의 영문표기】	PARK,Sung Jin
【주민등록번호】	740612-1051712
【우편번호】	140-120
【주소】	서울특별시 용산구 효창동 6-26 3/3
【국적】	KR
【발명자】	
【성명의 국문표기】	김민구
【성명의 영문표기】	KIM,Min Goo
【주민등록번호】	640820-1067025
【우편번호】	442-470
【주소】	경기도 수원시 팔달구 영통동 968 신나무실 신명아파트 633-1502
【국적】	KR
【발명자】	
【성명의 국문표기】	최순재
【성명의 영문표기】	CHOI,Soon Jae

**【주민등록번호】** 700114-1357114  
**【우편번호】** 449-907  
**【주소】** 경기도 용인시 기흥읍 신갈리 갈현마을 현대아파트  
 504-1202  
**【국적】** KR  
**【취지】** 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대  
 리인 이건  
 주 (인)  
**【수수료】**  

<b>【기본출원료】</b>	20 면	29,000 원
<b>【가산출원료】</b>	57 면	57,000 원
<b>【우선권주장료】</b>	0 건	0 원
<b>【심사청구료】</b>	0 항	0 원
<b>【합계】</b>		86,000 원

**【요약서】****【요약】**

통신시스템의 터보 복호화를 위한 장치가 개시되어 있다. 이러한 본 발명은 터보 복호기와 동일한 주파수에서 동작하는 고속 메모리 버퍼를 수신단의 메모리 버퍼와 터보 복호기의 사이에 접속하고, 상기 고속 메모리 버퍼를 통해 상기 수신단의 메모리 버퍼에 저장되어 있는 데이터 비트들을 읽어들이어 상기 터보 복호기에서 요구되는 시간만큼 지연시킨 후 상기 터보 복호기의 SISO 복호기로 입력하는 복호화 장치를 제안한다. 상기 수신단의 메모리 버퍼는 터보 복호기의 동작 주파수(클럭)에서 데이터 비트들을 출력한다. 이에 따라 본 발명은 터보 복호기에서 처리되어야 할 데이터의 속도가 증가하는 경우에도 상기 수신단의 메모리 버퍼의 동작 주파수를 증가시킬 필요성을 제거하는 것으로, 저 전력의 소모가 요구되는 이동통신 환경에 적합한 회로 구현을 가능하게 한다.

**【대표도】**

도 8

**【색인어】**

터보 복호기, 메모리 버퍼, 시소(SISO) 복호기

【명세서】

【발명의 명칭】

터보 복호화 장치 {TURBO DECODING APPARATUS}

【도면의 간단한 설명】

도 1은 일반적인 이동 통신시스템의 수신기 구성을 보여주는 도면.

도 2는 종래 기술에 따른 터보 복호화 장치의 구성을 보여주는 도면.

도 3은 종래 기술에 따른 시소(SISO) 복호기의 구성을 보여주는 도면.

도 4a 및 도 4b는 종래 기술에 따른 SISO 복호기에 의한 매트릭 연산 순서를 보여주는 도면.

도 5a 및 도 5b는 종래 기술에 따른 SISO 복호기에 의한 프레임 모드 및 윈도우 모드에서의 연산 순서를 보여주는 도면.

도 6은 도 3에 도시된 SISO 복호기에 의한 데이터 비트 입력 및 매트릭 출력의 처리 흐름을 보여주는 도면.

도 7은 도 3에 도시된 SISO 복호기에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면.

도 8은 본 발명의 실시예에 따른 SISO 복호기의 구성을 보여주는 도면.

도 9는 도 8에 도시된 고속 메모리 버퍼의 구성의 일 예를 보여주는 도면.

도 10은 도 9에 도시된 제어로직에 의한 제어 동작을 보여주는 도면.

도 11은 도 9에 도시된 고속 메모리 버퍼에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면.

도 12는 도 8에 도시된 고속 메모리 버퍼의 구성의 다른 예를 보여주는 도면.

도 13은 도 12에 도시된 제어로직에 의한 제어 동작을 보여주는 도면.

도 14는 도 12에 도시된 (알파) 시프트 레지스터에 의한 데이터 처리 흐름을 보여주는 도면.

도 15는 도 12에 도시된 (베타) 시프트 레지스터에 대한 데이터 처리 흐름을 보여주는 도면.

도 16은 도 12에 도시된 고속 메모리 버퍼에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면.

도 17은 도 8에 도시된 고속 메모리 버퍼의 구성의 또 다른 예를 보여주는 도면.

도 18은 도 17에 도시된 제어로직에 의한 제어 동작을 보여주는 도면.

도 19는 도 17에 도시된 (알파) 시프트 레지스터에 의한 데이터 처리 흐름을 보여주는 도면.

도 20은 도 17에 도시된 고속 메모리 버퍼에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면.

## 【발명의 상세한 설명】

## 【발명의 목적】

## 【발명이 속하는 기술분야 및 그 분야의 종래기술】

- <21> 본 발명은 통신시스템의 복호화에 관한 것으로, 특히 터보 복호화를 위한 장치에 관한 것이다.
- <22> 일반적으로 디지털 통신 시스템에서는 데이터 전송시 채널상에서 발생할 수 있는 오류를 효과적으로 정정하여 데이터 전송의 신뢰도를 높이기 위하여 순방향 오류 정정(FEC: Forward Error Correction) 부호가 널리 사용되고 있다. 이러한 순방향 오류 정정 부호로 터보 부호(turbo code)가 있다. 터보 부호는 길쌈 부호(convolution code)에 비해 데이터의 고속 전송시 오류 정정 능력이 뛰어나기 때문에, 제3 세대 이동 통신시스템으로서 주목받고 있는 동기 방식의 cdma2000(code division multiple access 2000) 시스템과 비동기 방식의 UMTS(Universal Mobile Telecommunication System) 등에 모두 채택되었다.
- <23> 도 1은 제3 세대 이동 통신시스템의 수신기에 대한 구성을 보여주는 도면이다. 이 구성은 고속의 패킷 데이터 통신을 가능하게 하는 1x EV-DV(Evolution Data and Voice) 시스템의 수신기에 대한 구성을 예로서 나타내고 있다.
- <24> 상기 도 1에서, 수신 신호는 수신신호 처리부 10에 의해 고주파수(RF: Radio Frequency), 중간주파수(IF: Intermediate Frequency) 및 기저대역(baseband) 처리된다. 상기 수신신호 처리부 10에 의해 처리된 신호는 각 채널별로 분리된다. 수신기 30은 순방향 기본채널(F-FCH: Forward Fundamental Channel), 순방향 부가채널(F-SCH: Forward



Supplemental Channel), 순방향 전용제어채널(F-DCCH: Forward Dedicated Control Channel)의 신호들을 수신하여 처리한다. 수신기 40은 순방향 패킷데이터 채널(F-PDCH: Forward Packet Data Channel) 신호를 수신하여 처리한다. 수신기 50은 순방향 패킷데이터 제어채널(F-PDCCH: Forward Packet Data Control Channel) 신호를 수신하여 처리한다. 상기 수신기 40은 블록들 42,44,46,48을 포함하여 구성된다. 상기 블록 42는 채널상에서 발생할 수 있는 손실을 최소화하는 기능을 위한 것으로, 핑거(finger) 및 결합기(combiner)를 포함한다. 상기 블록 44는 채널 복호화가 가능하도록 신호를 변환하는 기능을 위한 것으로, 복조기 버퍼(demodulation buffer), 왈시 역확산기(walsh decoder), 심볼 복조기(symbol demapper), 디스크램블러(descrambler)를 포함한다. 상기 블록 48은 복호화 및 복호화된 결과를 L1 계층(Layer) 처리부 70이 참조할 수 있도록 하는 기능을 위한 것으로, 터보 복호기(turbo decoder) 및 출력 버퍼를 포함한다. 상기 블록 46은 복조 심볼을 복호화를 위해 상기 블록 48로 전달하는 기능을 위한 것으로, 결합기(combiner), 역혼합기(deshuffler), 디인터리버(deinterleaver), 메모리 버퍼(memory buffer)를 포함한다.

<25> 위에서 미설명된 탐색기(searcher) 20은 수신신호의 탐색을 위한 구성 요소이고, HARQ(Hybrid Automatic Request) 제어기 60은 수신되지 않은 심볼에 대한 재전송을 요청하기 위한 구성 요소이다.

<26> 도 2는 종래 기술에 따른 터보 복호화 장치의 구성을 보여주는 도면으로, 도 1에 도시된 터보 복호기 블록 48의 구성에 해당한다. 이 도면은 터보 복호기가 시소(SISO: Soft-In Soft-Out) 방식으로 구성된 예를 나타낸다. SISO 방식을 대신하여, 상기 터보 복호기는 MAP(Maximum A Posteriori)방식 또는 RESOVA(Register

Exchange Soft Output Viterbi Algorithm) 방식으로 구현될 수도 있다. 여기서, SISO 방식은 심볼에 대한 신뢰도를 가지고 확률을 구하는 방식이고, RESOVA 방식은 심볼이 거쳐 가는 경로(path)를 기다란 부호어(codeword)라고 생각하고 부호어에 대한 확률을 구하는 방식이다.

<27>       상기 도 2를 참조하면, 상기 블록 48의 입력단으로는 도 1에 도시된 블록 46의 메모리 버퍼 46-1에 저장되어 있는 심볼들(데이터 비트들)이 제공된다. 상기 메모리 버퍼 46-1에는 디인터리빙이 완료된 비트들이 구조적 부호인 시스템 부호(systematic code)와, 비구조적 부호인 패리티 부호들(parity 1 code, parity 2 code)로 구분되어 저장된다. 이때 상기 메모리 버퍼 46-1로부터 상기 블록 48로는 시스템 부호의 비트들과, 패리티 부호들의 비트들이 동시에 제공된다. 예컨대, 1x EV-DV 시스템에서 상기 메모리 버퍼 46-1은 송신단에서 준보완터보부호(QCTC: Quasi-Complementary Turbo Code)로 부호화 및 송신된 후 수신되는 심볼들을 저장하기 위한 QCTC 메모리 버퍼이다. 이 메모리 버퍼 46-1에서 출력되는 하나의 부호는 M 비트로 구성되고 상기 메모리 버퍼 46-1로부터는 구조적 부호와 비구조적 부호의 3개의 부호들이 모두 출력되기 때문에, 상기 메모리 버퍼 46-1과 상기 블록 48의 사이에는 (3 채널) 비트의 버스가 구성되며 상기 메모리 버퍼 46-1로부터 출력되는 부호들은 상기 블록 48의 다중화기(MUX: Multiplexer) 48-1로 제공된다.

<28>       상기 터보 복호기 블록 48은 다중화기 48-1, SISO 방식 복호기(이하 "SISO 복호기"라 칭함) 48-2, 인터리버(interleaver) 48-3, 디인터리버(deinterleaver) 48-4, 출력 버퍼(output buffer) 48-5, CRC 검사기(Cyclic Redundancy Code

checker) 48-6으로 이루어진다. 상기 다중화기 48-1은 상기 메모리 버퍼 46-1로부터의 비트들과, 상기 인터리버 48-3의 출력과, 상기 디인터리버 48-4의 출력을 입력하고, 다중화하여 출력한다. 상기 SISO 복호기 48-2는 상기 다중화기 48-1의 출력을 입력하고, 후술될 도 3에 도시된 바와 같은 구성하에 SISO 복호화하여 출력한다. 상기 인터리버 48-3은 상기 SISO 복호기 48-2의 출력을 인터리빙한다. 상기 디인터리버 48-4는 상기 SISO 복호기 48-2의 출력을 디인터리빙한다. 상기 출력 버퍼 48-5는 상기 디인터리버 48-4에 의해 디인터리빙된 결과를 L1 계층(Layer) 처리부 70이 참조할 수 있도록 저장한다. CRC 검사기 48-6은 상기 디인터리버 48-4에 의한 디인터리빙 결과에 대해 CRC 검사를 하고, 그 검사 결과를 상기 L1 계층 처리부 70으로 제공한다.

<29>        도 3은 종래 기술에 따른 SISO 복호기의 구성을 보여주는 도면이다. 이 도면은 SISO 복호기가 윈도우 모드(sliding window mode) 방식으로 구현된 예를 나타내는 것으로, 여기서는 윈도우 개수가 2인 것으로 가정되고 있다. SISO 복호기는 MAP 방식의 복호기와 비교하였을 때 기본 구조는 동일하고 출력 값에서만 차이가 있다.

<30>        상기 도 3을 참조하면, SISO 복호기가 복호화하는 과정에서는 몇가지 메트릭들(metrics)을 연산하는 동작이 수행된다. 즉, SISO 복호기의 복호화 동작중에 델타 메트릭(delta metric), 알파 메트릭( $\alpha$ metric), 베타 메트릭( $\beta$ metric) 및 로그우도비율(LLR: Log Likelihood Ratio)의 값들이 계산된다. 역다중화기(DEMUX: Demultiplexer) 205는 미리 설정된 속도 - 예를 들어, 터보 복호기의 클럭(동작 주파수)의 3배 속도로 메모리 버퍼 46-1에 저장되어 있는 데이터 비트들을

엑세스(access)하고, 제1 출력(①), 제2 출력(②) 및 제3 출력(③)을 제공한다. 델타 메트릭 계산부 210은 3개의 계산기들 211~213으로 구성되고, 각 계산기들은 상기 제1 출력, 제2 출력 및 제3 출력 각각에 대한 델타 메트릭들을 계산한다. 알파 메트릭 계산부 220은 델타 메트릭 계산기 211에 의해 계산된 델타 메트릭을 입력하고, 해당하는 알파 메트릭을 계산한다. 베타 메트릭 계산부 230은 2개의 계산기들 231, 232와, 다중화기 233으로 구성된다. 즉, 상기 베타 메트릭 계산부 230은 제1 베타( $\beta 1$ ) 메트릭을 계산하는 계산기 231과, 제2 베타( $\beta 2$ ) 메트릭을 계산하는 계산하는 계산기 232와, 상기 계산기들 231, 232에 의한 계산 결과들을 다중화하는 다중화기 233을 포함한다. LLR 계산부 240은 3개의 계산기들 241~243으로 구성되고, 상기 알파 메트릭 계산부 220에 의해 계산된 알파 메트릭과 상기 다중화기 233에 의한 다중화 결과를 입력하여 해당하는 LLR의 값들을 계산한다. 감산부 250은 3개의 감산기들 251~253으로 구성되고, 각 감산기들은 상기 LLR 계산기들 241~243에 의해 계산된 LLR의 값들로부터 상기 역다중화기 205의 제1 출력(①)을 감산하고 그 감산 결과들을 인터리빙/디인터리빙을 위해 도 2에 도시된 인터리버 48-3과 디인터리버 48-4로 출력한다.

<31>        전술한 바와 같이 종래 기술에 따른 SISO 복호기는 메트릭 연산을 위한 델타 메트릭 계산부, 알파 메트릭 계산부 및 베타 메트릭 계산부와, 확률을 근거로 복호화하여 결과를 출력하는 LLR 계산부로 구성된다. 여기서, 베타 메트릭 계산부는 윈도우의 수에 따라 2개의 계산기들로 구성된 것으로 나타내고 있다.

<32>        상기 델타 메트릭(Delta metric)은 스테이트 메트릭(state metric)이라고도

하며 부호기의 한 스테이트에서 다른 스테이트로의 천이 확률을 나타낸다. 상기 알파( $\alpha$ ) 메트릭은 순방향 스테이트 메트릭(forward state metric)이라고도 하며, 이전 스테이트에서 다음 스테이트로의 스테이트 메트릭과 이전 스테이트가 될 확률을 더한 값을 나타낸다. 상기  $\alpha$  메트릭은 처음의 수신 신호부터 계산하고 있는 신호의 구간에 걸쳐서 누적된 확률을 뜻하며 순차적으로 계산된다. 상기 베타( $\beta$ ) 메트릭은 역방향 스테이트 메트릭(backward state metric)이라고도 하며, 현재 스테이트에서 이전 스테이트로의 누적 확률을 나타낸다. 상기  $\alpha$  메트릭과  $\beta$  메트릭이 모두 구해지면, LLR의 값이 계산된다. LLR은 심볼에 대한 확률을 나타내는 것으로, "1"이 될 확률과 "0"이 될 확률의 비율을 로그 스케일(log scale)로 표현한 값이다. LLR을 계산하기 위한 LLR 계산기들 241~243 각각은 순방향 스테이트, 역방향 스테이트에 대한 천이 확률을 근거로 심볼에 대한 확률을 구한다. 이때 LLR의 값이 양수일 경우 심볼 "1", 음수일 경우 심볼 "0"을 나타낸다. 이와 같이 수신된 신호를 복호화하기 위해서 SISO 복호기는  $\alpha$  메트릭 및  $\beta$  메트릭의 값을 모두 계산한다. 여기서  $\beta$  메트릭은 메모리 버퍼 46-1에 저장된 수신 신호의 역순으로 값을 연산해야 하기 때문에,  $\beta$  메트릭의 연산이 모두 끝나기 전까지는 LLR의 값을 구할 수 없다는 사실에 유의할 필요가 있다.

<33> 도 4는 종래 기술에 따른 SISO 복호기에 의한 메트릭 연산 순서를 보여주는 도면으로, 도 4a와 도 4b로 구성된다. 상기 도 4a는  $\alpha$  메트릭을 연산하는 과정을 보여주고, 상기 도 4b는  $\beta$  메트릭을 연산하는 과정을 보여준다. 상기 도 4a 및 도 4b를 참조하면,  $\alpha$  메트릭을 연산하는 과정과,  $\beta$  메트릭을 연산하는 과정 사이에는 차이가 있음을 알 수 있다.  $\alpha$  메트릭  $\alpha_k$  는 이전 값인  $(k-1)$ 번째  $\alpha$  메트릭으로부터 계산되며,  $\beta$  메트릭  $\beta_k$  는 나중 값인  $(k+1)$ 번째  $\beta$  메트릭으로부터 계산된다. 이와 같이  $\beta$  메트릭을 연산하기 위해

서는 수신된 신호를 역순으로 참조하여야 하기 때문에 전체 수신 신호의 길이만큼의 초기 지연이 발생하게 된다.

- <34> 도 5는 종래 기술에 따른 SISO 복호기에 의한 프레임 모드 및 윈도우 모드에서의 연산 순서를 보여주는 도면으로, 도 5a와 도 5b로 구성된다. 상기 도 5a는 SISO 복호기가 프레임 모드(frame mode)에서 메트릭들을 연산하는 순서를 나타내고, 상기 도 5b는 SISO 복호기가 도 3에 도시된 바와 같이 윈도우 모드(window mode)에서 메트릭들을 연산하는 순서를 나타낸다.
- <35> 상기 도 5a를 참조하면,  $\beta$  메트릭이 모두 계산된 이후에  $\alpha$  메트릭과 LLR의 값( $\lambda$ )이 계산되기 때문에, 프레임의 구간만큼의 초기 지연이 발생된다. 이와 같이 프레임 모드 방식의 SISO 복호기는  $\beta$  메트릭을 계산한 다음에  $\alpha$  메트릭을 계산하여 LLR의 값( $\lambda$ ) 값을 계산한다. 그러므로  $\beta$  메트릭을 계산하는 동안의 지연 시간이 발생하는데, 이러한 초기 지연을 줄이기 위해 제안된 것이 윈도우 모드의 방식이다.
- <36> 상기 도 5b를 참조하면, 윈도우 모드의 SISO 복호기는  $\beta$  메트릭 계산을 위해 수신된 신호를 임의의 길이로 분할한다. 임의의 길이로 분할된 수신 신호를 가지고  $\beta$  메트릭을 계산하면 초기에 계산되는 값들은 정확하지 않은 확률을 가지지만, 나중에 갈수록 더 정확한 값들이 계산되어진다. 실제 LLR을 계산하는데는 이러한 정

확한 값이 나오는 구간부터 계산된 값을 사용하면 된다. 이때 계산의 편리성을 위해 부정확한 구간과 신뢰할 수 있는 구간의 크기는 같게 설정된다. 하나의 윈도우가 정확한 값들을 연산하는 동안 다른 윈도우는 부정확한 값들을 연산하여 서로 맞물리게끔 구성하는 것이다. 이와 같이  $\beta$  메트릭을 계산함에 있어서 2가지의 윈도우를 사용한 예가 전술한 도 3에 도시된 베타 메트릭 계산부 230이다. 그러므로, 윈도우 모드의 SISO 복호기는  $\alpha$  메트릭과,  $\beta 1$  메트릭과,  $\beta 2$  메트릭의 3개의 값들을 계산한다. 이러한 3개의 메트릭들을 계산함에 있어서는 델타 메트릭의 계산이 선행된다.

<37> 다시 도 3을 참조하면, 델타 메트릭 계산기들 211~213은 각각 메모리 버퍼 46-1의 서로 다른 주소에 저장되어 있는 수신 신호의 데이터 비트들을 입력하여 해당하는 델타 메트릭을 계산한다. 즉, 델타 메트릭 계산기들 211~213 각각은 도 7에 도시된 바와 같이 터보 복호기의 동작 주파수 1 클럭의 시간동안 메모리 버퍼 46-1로부터 서로 다른 위치의 신호들을 읽어들인다.

<38> 도 6은 도 3에 도시된 SISO 복호기에 의한 데이터 비트 입력 및 메트릭 출력의 처리 흐름을 보여주는 도면이다.

<39> 상기 도 6을 참조하면, 상기 SISO 복호기의 델타 메트릭 계산기들 211~213에는 각각 메모리 버퍼 46-1의 서로 다른 주소에 저장되어 있는 수신 신호의 데이터 비트들이 인가됨을 알 수 있다. 가로는 시간 축을 나타내는데, 시간이 경과함에 따라 상기 델타 메트릭 계산기들 211~213에는 서로 다른 데이터의 비트들이 제공됨을 알 수 있다. 이러한 동작을 위해서는 상기 메모리 버퍼 46-1이 터보 복호기의 동작 주파수보다 3배 빠르게 액세스되어야 한다. 즉, 상기 메모리 버퍼 46-1의 클럭으로는 터보 복호기 클럭의 3배에 해당하는 클럭이 사용되어야 한다.

- <40> 도 7은 도 3에 도시된 SISO 복호기에 의한 메모리 버퍼 액세스 동작의 타이밍을 보여주는 도면이다.
- <41> 상기 도 7을 참조하면, SISO 복호기는 도 3의 메모리 버퍼 46-1의 서로 다른 주소들 addr1, addr2 및 addr3에 저장되어 있는 데이터 비트들 data1, data2 및 data3을 가져와서  $\alpha$  메트릭을 위한 델타 메트릭,  $\beta_1$ 을 위한 델타 메트릭 및  $\beta_2$ 를 위한 델타 메트릭을 계산한다. 이를 위해 메모리 버퍼 46-1의 읽기 동작은 터보 복호기 클럭의 3배 속도로 이루어진다.
- <42> 상기 도 6 및 도 7에 도시된 메모리 버퍼 액세스 및 데이터 처리의 동작은 윈도우 크기 W가 실제 적용되는 길이보다 훨씬 짧은 W=4인 것으로 가정한 경우이다. 실제 고속의 터보 복호기에 적용될 때에 윈도우 크기 W = 24~48 정도로 설정될 것이며 상황에 따라 더 길게 설정될 수도 있을 것이다. 윈도우 크기 W가 변화함에 따른 버퍼의 구조 자체는 변화하지 않으며 데이터 흐름도의 전체적인 모양도 변화하지 않고 길이의 비율 그대로 커진다.
- <43> 다시 도 6을 참조하면, 델타 블록 입력(delta block input)의 각 칸의 내부에 쓰여진 알파벳은 메모리 버퍼 46-1의 각기 다른 주소에 저장되어 있는 데이터 비트들로 델타 메트릭 계산부 210에 입력되는 값을 의미한다.  $\alpha$  메트릭에 비해  $\beta$  메트릭이 먼저 구해지며, 두개의  $\beta$  메트릭 계산기들 231, 232는 서로 어긋나서 동작한다(T1구간 및 T2구간 참조). 신뢰성 있는  $\beta_1$  메트릭이 출력되는 시점부터  $\alpha$  메트릭이 동시에 계산된다(T2구간 참조).  $\beta$  메트릭이 계산될 때 처음 W구간 동안은 부정확한 확률의 값들이 출력되지만 그 이후부터 W구간 동안에는 신뢰할 수 있는 확률의 메트릭 값이 출력된다.  $\alpha$  출력(out),  $\beta_1$  출력,  $\beta_2$  출력에서 칸 내부의 알파벳은 메트릭의 순서를 의미한다.  $\beta_1$  및



$\beta_2$ 를 위한 델타 메트릭 계산기들 212, 213의 출력들은 서로 겹치지 않고 어긋나기 때문에, 베타 메트릭 계산기들 231, 232에 의해 계산된 후 출력되는  $\beta$  메트릭은 연속적이다. 상기 도 6에는 점선으로 표시된 원 부분이 있는데, 이 부분은 그 시점에서 필요한 데이터 비트가 메모리 버퍼 46-1의 서로 다른 주소인 d, n, f 위치에 있는 수신 신호라는 것을 나타낸다.

- <44> 한편, 상기 도 3에 도시된 바와 같은 SISO 복호기를 고속의 전송 속도가 요구되는 1x EV-DV 시스템에 사용한다고 가정할 때, 대략 30~60MHz 정도의 주파수에서 동작하는 터보 복호기가 요구된다. 그러므로, 메모리 버퍼 46-1의 동작 주파수는 터보 복호기의 동작 주파수의 3배에 해당하는 90~180MHz 범위내에서 결정되어야 한다. 이러한 터보 복호기의 동작 주파수는 저전력의 소모가 요구되는 이동 통신 단말기에는 적합하지 않다.
- <45> 전술한 바와 같이, 대표적인 제3 세대 이동 통신시스템인 1x EV-DV와 같은 시스템은 고속의 패킷 데이터 통신을 가능하게 한다. 이러한 통신 시스템에서는 높은 성능을 위해 고속의 터보 복호기가 요구된다. 고속의 복호를 위해서는 터보 복호기의 전단에 연결된 메모리 버퍼에 저장되어 있는 데이터 비트들(심볼들)이 적절한 방식으로 터보 복호기로 입력되어야 한다. 윈도우 모드(sliding window) 방식의 SISO 복호기는 프레임 모드 방식의 SISO 복호기에 비해 초기 지연을 줄일 수 있기

때문에, 윈도우 모드 방식의 SISO 복호기를 터보 복호기에 사용하는 것이 바람직하다. 이때 윈도우 모드 방식의 SISO 복호기는 메모리 버퍼로부터 윈도우의 수에 대응하는 데이터 비트들을 읽어들인 후 복호화 동작을 수행한다. 예를 들어, 윈도우의 수가 2개인 경우 SISO 복호기는 메모리 버퍼로부터 3번에 걸쳐 데이터 비트들을 가져온 후 복호화를 위한 메트릭들을 계산한다. 이러한 동작은 터보 복호기가 저속으로 동작하는 경우에는 별다른 문제가 없지만, 고속으로 동작하는 경우에는 문제가 될 수 있다. 왜냐하면, 메모리 버퍼는 터보 복호기보다 3배 빠르게 동작되어야 하며 터보 복호기의 동작 주파수가 낮을 경우에 그 동작 주파수보다 3배 빠른 메모리 버퍼를 사용하는 것이 이동통신 단말기에 별다른 무리가 되지 않지만, 터보 복호기의 동작 주파수가 높을 경우에는 그 동작 주파수보다 3배 빠른 메모리 버퍼를 사용하는 것이 이동통신 단말기에 상당한 무리가 될 것이기 때문이다. 예컨대, 고속 데이터 서비스의 제공을 목적으로 하는 cdma2000나 UMTS의 터보 복호기는 고속으로 동작하여야만 제 성능을 발휘할 수 있는데, 이에 따라 메모리 버퍼의 동작 주파수도 증가시키면 이동통신 단말기에서 소모되는 전력량은 급증하게 될 것이다. 이러한 전력 소모량의 급증은 저전력 설계가 요구되는 이동통신 단말기에는 적합하지 않다.

#### 【발명이 이루고자 하는 기술적 과제】

- <46> 따라서 본 발명의 목적은 1x EV-DV 시스템과 같이 고속의 패킷 데이터를 서비스하는 통신시스템에 사용하기 위한 터보 복호화 장치를 제공함에 있다.
- <47> 본 발명의 다른 목적은 고속으로 동작하는 이동 통신단말기에서 복호화를 위한 터보 복호기의 동작 주파수와 복호화를 위한 데이터 비트들을 상기 터보 복호기로 인가하

는 메모리 버퍼의 동작 주파수를 동일하게 사용 가능하도록 하는 장치를 제공함에 있다.

<48> 본 발명의 또 다른 목적은 고속으로 동작하는 이동 통신단말기에서 복화화를 위한 수신 데이터 비트들을 저장하고 있는 메모리 버퍼의 동작 주파수를 고속으로 증가시킬 필요성을 제거함으로써 저전력의 소모가 요구되는 이동 통신단말기 구현을 가능하게 하는 장치를 제공함에 있다.

<49> 이러한 목적들을 달성하기 위한 본 발명은 터보 복호기와 동일한 주파수에서 동작하는 고속 메모리 버퍼를 수신단의 메모리 버퍼와 터보 복호기의 사이에 접속하고, 상기 고속 메모리 버퍼를 통해 상기 수신단의 메모리 버퍼에 저장되어 있는 데이터 비트들을 읽어들이어 상기 터보 복호기에서 요구되는 시간만큼 지연시킨 후 상기 터보 복호기의 SISO 복호기로 입력하는 복호화 장치를 제안한다. 상기 수신단의 메모리 버퍼는 터보 복호기의 동작 주파수(클럭)에서 데이터 비트들을 출력한다. 이에 따라 본 발명은 터보 복호기에서 처리되어야 할 데이터의 속도가 증가하는 경우에도 상기 수신단의 메모리 버퍼의 동작 주파수를 증가시킬 필요성을 제거하는 것으로, 저전력의 소모가 요구되는 이동 통신 환경에 적합한 회로 구현을 가능하게 한다.

<50> 본 발명의 제1 견지(aspect)에 따르면, 통신시스템의 터보 복호화 장치는, 메모리 버퍼와, SISO 복호기를 포함한다. 상기 메모리 버퍼는 단방향 시프트 레지스터와, 적어도 하나 이상의 양방향 시프트 레지스터들로 구성된다. 상기 SISO 복호기는 제1 메트릭 계산부 내지 제2 메트릭 계산부와, 감산부로 구성된다.

<51> 상기 단방향 시프트 레지스터는, 데이터 입력을 위한 입력 단자와 데이터 출력을 위한 출력 단자를 구비하고, 입력 데이터 비트들을 상기 입력 단자를 통해 순차적으로

입력 및 시프트시켜 제1 길이의 비트 열들을 형성한 후 상기 제1 길이의 비트 열들을 상기 출력 단자를 통해 순차적으로 출력한다.

<52>       상기 적어도 하나 이상의 양방향 시프트 레지스터들 각각은, 데이터 입/출력을 위한 제1 단자 및 제2 단자를 구비하고, 상기 입력 데이터 비트들은 상기 제1 길이의 1/2 인 제2 길이의 비트들로 이루어지는 그룹들로 분할되고, 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 순차적으로 입력 및 시프트시켜 상기 제2 길이의 비트 열들을 형성한 후 상기 제1 단자를 통해 순차적으로 출력하고, 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 순차적으로 입력 및 시프트시켜 상기 제2 길이의 비트 열들을 형성한 후 상기 제2 단자를 통해 순차적으로 출력한다.

<53>       상기 제1 메트릭 계산부는, 상기 각 시프트 레지스터들로부터의 출력 비트들을 입력하고 해당하는 델타 메트릭들을 각각 계산한다. 상기 제2 메트릭 계산부는, 상기 단방향 시프트 레지스터에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭을 입력하여 알파 메트릭을 계산한다. 상기 제3 메트릭 계산부는, 상기 양방향 시프트 레지스터들 각각에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭들을 입력하여 베타 메트릭들을 각각 계산한다. 상기 제4 메트릭 계산부는, 상기 알파 메트릭을 입력하고, 또한 상기 베타 메트릭들의 다중화 결과를 입력하여, 상기 각 시프트 레지스터들에 대응하는 로그우도비율(LLR) 값들을 계산한다. 상기 감산부는, 상기 각 LLR 값으로부터 상기 단방향 시프트 레지스터의 출력을 각각 감산하고, 감산 결과들을 인터리빙/디인터리빙을 위해 출력한다.

- <54> 바람직하기로, 상기 메모리 버퍼는, 상기 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 입력 데이터 비트들을 상기 제1 단자 또는 상기 제2 단자로 인가되도록 제어하기 위한 선택신호들을 상기 양방향 시프트 레지스터들 각각에 제공하는 제어 로직을 더 포함한다.
- <55> 바람직하기로, 상기 메모리 버퍼는, 상기 양방향 시프트 레지스터들 각각에 대응하는 역다중화기와, 다중화기를 더 포함한다. 상기 역다중화기는, 상기 입력 데이터 비트들을 입력하기 위한 입력단과, 상기 제1 단자 및 상기 제2 단자에 각각 접속되는 제1 출력단 및 제2 출력단을 구비하고, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 홀수번째 그룹들의 비트들을 상기 제1 출력단을 통해 상기 제1 단자로 인가하고, 상기 짝수번째 그룹들의 비트들을 상기 제2 출력단을 통해 상기 제2 단자로 인가한다. 상기 다중화기는, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제1 단자를 통해 출력되는 비트열들과 상기 제2 단자를 통해 출력되는 비트열들을 다중화하여 상기 제1 메트릭 계산부로 출력한다.
- <56> 바람직하기로, 상기 선택신호들 각각은 상기 입력 데이터 비트들이 상기 양방향 시프트 레지스터들 각각에 서로 다른 시점에서 인가되도록 제어하기 위한 신호들이다.
- <57> 바람직하기로, 상기 홀수번째 그룹들의 비트들이 상기 제1 단자를 통해 순차적으로 출력됨과 동시에 상기 짝수번째 그룹들의 비트들이 상기 제2 단자를 통해 순차적으로 입력 및 시프트된다.
- <58> 바람직하기로, 상기 양방향 시프트 레지스터들의 수는 윈도우 수에 의해 결정된다.

- <59> 바람직하기로, 상기 제1 길이 및 상기 제2 길이는 윈도우 사이즈와 윈도우 수에 의해 결정된다.
- <60> 바람직하기로, 상기 제2 길이는 상기 윈도우 사이즈와 상기 윈도우 수의 곱에 의해 결정된다.
- <61> 바람직하기로, 상기 입력 데이터 비트들은 터보 복호기의 클럭 속도로 입력된다.
- <62> 본 발명의 제2 견지에 따르면, 통신시스템의 터보 복호화 장치는 메모리 버퍼와, SISO 복호기를 포함한다. 상기 메모리 버퍼는 제1 스테이지의 양방향 시프트 레지스터들과, 제2 스테이지의 양방향 시프트 레지스터로 구성된다. 상기 SISO 복호기는 제1 메트릭 계산부 내지 제4 메트릭 계산부와, 감산부로 구성된다.
- <63> 상기 제1 스테이지의 양방향 시프트 레지스터들 각각은, 데이터 입/출력을 위한 제1 단자 및 제2 단자를 구비하고, 입력 데이터 비트들은 미리 설정된 길이의 비트들로 이루어지는 그룹들로 분할되고, 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제1 단자를 통해 순차적으로 출력하고, 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제2 단자를 통해 순차적으로 출력한다.
- <64> 상기 제2 스테이지의 양방향 시프트 레지스터는, 데이터 입/출력을 위한 제3 단자 및 제4 단자를 구비하고, 상기 제1 단자를 통해 순차적으로 출력되는 비트들을 상기 제3 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제3 단자를 통해 순차적으로 출력하고, 상기 제2 단자를 통해 순차적으로 출력되는 비트

들을 상기 제4 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제4 단자를 통해 순차적으로 출력한다.

<65>       상기 제1 메트릭 계산부는, 상기 각 시프트 레지스터들로부터의 출력 비트들을 입력하고 해당하는 델타 메트릭들을 각각 계산한다. 상기 제2 메트릭 계산부는, 상기 단방향 시프트 레지스터에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭을 입력하여 알파 메트릭을 계산한다. 상기 제3 메트릭 계산부는, 상기 양방향 시프트 레지스터들 각각에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭들을 입력하여 베타 메트릭들을 각각 계산한다. 상기 제4 메트릭 계산부는, 상기 알파 메트릭을 입력하고, 또한 상기 베타 메트릭들의 다중화 결과를 입력하여, 상기 각 시프트 레지스터들에 대응하는 로그우도비율(LLR) 값들을 계산한다. 상기 감산부는, 상기 각 LLR 값으로부터 상기 단방향 시프트 레지스터의 출력을 각각 감산하고, 감산 결과들을 인터리빙/디인터리빙을 위해 출력한다.

<66>       바람직하기로, 상기 메모리 버퍼는, 상기 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 입력 데이터 비트들을 상기 제1 단자 또는 상기 제2 단자로 인가되도록 제어하기 위한 선택신호들을 상기 제1 스테이지의 양방향 시프트 레지스터들 각각에 제공하는 제어 로직을 더 포함한다.

<67>       바람직하기로, 상기 메모리 버퍼는, 상기 제1 스테이지의 양방향 시프트 레지스터들 각각에 대응하는 역다중화기와, 다중화기를 더 포함한다. 상기 역다중화기는, 상기 입력 데이터 비트들을 입력하기 위한 입력단과, 상기 제1 단자 및 상기 제2 단자에 각각 접속되는 제1 출력단 및 제2 출력단을 구비하고, 상기 제어 로직으로부터 제공되는 해당

하는 선택신호에 응답하여 상기 홀수번째 그룹들의 비트들을 상기 제1 출력단을 통해 상기 제1 단자로 인가하고, 상기 짝수번째 그룹들의 비트들을 상기 제2 출력단을 통해 상기 제2 단자로 인가한다. 상기 다중화기는, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제1 단자를 통해 출력되는 비트들과 상기 제2 단자를 통해 출력되는 비트들을 다중화하여 상기 제1 메트릭 계산부로 출력한다.

<68> 바람직하기로, 상기 메모리 버퍼는, 상기 제2 스테이지의 양방향 시프트 레지스터에 대응하고, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제3 단자를 통해 출력되는 비트들과 상기 제4 단자를 통해 출력되는 비트들을 다중화하여 상기 제1 메트릭 계산부로 출력하는 다중화기를 더 포함한다.

<69> 바람직하기로, 상기 선택신호들 각각은 상기 입력 데이터 비트들이 상기 양방향 시프트 레지스터들 각각에 서로 다른 시점에서 인가되도록 제어하기 위한 신호들이다.

<70> 바람직하기로, 상기 홀수번째 그룹들의 비트들이 상기 제1 단자를 통해 순차적으로 출력됨과 동시에 상기 짝수번째 그룹들의 비트들이 상기 제2 단자를 통해 순차적으로 입력 및 시프트된다.

<71> 바람직하기로, 상기 제1 스테이지의 양방향 시프트 레지스터들의 수는 윈도우 수에 의해 결정된다.

<72> 바람직하기로, 상기 제1 길이 및 상기 제2 길이는 윈도우 사이즈와 윈도우 수에 의해 결정된다.

<73> 바람직하기로, 상기 제2 길이는 상기 윈도우 사이즈와 상기 윈도우 수의 곱에 의해 결정된다.



- <74> 바람직하기로, 상기 입력 데이터 비트들은 터보 복호기의 클럭 속도로 입력된다.
- <75> 전술한 바와 같은 내용은 당해 분야 통상의 지식을 가진 자는 후술되는 본 발명의 구체적인 설명으로 보다 잘 이해할 수 있도록 하기 위하여 본 발명의 특징들 및 기술적인 장점들을 다소 넓게 약술한 것이다.
- <76> 본 발명의 청구범위의 주제를 형성하는 본 발명의 추가적인 특징들 및 장점들이 후술될 것이다. 당해 분야에서 통상의 지식을 가진 자는 본 발명의 동일한 목적들을 달성하기 위하여 다른 구조들을 변경하거나 설계하는 기초로서 발명의 개시된 개념 및 구체적인 실시 예가 용이하게 사용될 수도 있다는 사실을 인식하여야 한다. 당해 분야에서 통상의 지식을 가진 자는 또한 발명과 균등한 구조들이 본 발명의 가장 넓은 형태의 사상 및 범위로부터 벗어나지 않는다는 사실을 인식하여야 한다.

#### 【발명의 구성 및 작용】

- <77> 이하 본 발명의 바람직한 실시예의 상세한 설명이 첨부된 도면들을 참조하여 설명될 것이다. 도면들 중 참조번호들 및 동일한 구성요소들에 대해서는 비록 다른 도면상에 표시되더라도 가능한 한 동일한 참조번호들 및 부호들로 나타내고 있음에 유의해야 한다. 하기에서 본 발명을 설명함에 있어, 관련된 공지 기능 또는 구성에 대한 구체적인 설명이 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략할 것이다.
- <78> 도 8은 본 발명의 실시예에 따른 복호화 장치의 구성을 보여주는 도면이다. 이 도면은 복호기가 도 2에 도시된 바와 같이 터보 복호기로 구성되는 경우에, 그 터보 복호

기를 구성하는 SISO 복호기와 이 SISO 복호기의 전단에 연결되는 메모리 버퍼의 구조만을 나타내고 있다.

<79>        상기 도 8을 참조하면, 본 발명의 실시예에 따른 복호화 장치는 도 3에 도시된 바와 같이 델타 메트릭 계산부 210과, 알파 메트릭 계산부 220과, 베타 메트릭 계산부 230과, LLR 계산부 240과, 감산부 250으로 이루어지는 SISO 복호기를 포함한다. 본 발명의 실시예에 따른 복호화 장치는 이러한 SISO 복호기의 전단에 상기 도 3에 도시된 바와 같이 수신 심볼들(데이터 비트들)을 저장하고 있는 메모리 버퍼 46-10을 구비할 뿐만 아니라, 이 메모리 버퍼 46-10과 SISO 복호기의 사이에 접속된 고속 메모리 버퍼(High Rate Memory Buffer) 260을 더 구비하는 것을 특징으로 한다. 게다가, 본 발명의 실시예에 따른 복호화 장치의 메모리 버퍼 46-10은 종래 기술에 따른 복호화 장치의 메모리 버퍼 46-1과 같이 {터보 복호기의 클럭  $\times 8$ }의 속도로 동작하는 것이 아니라, 터보 복호기의 1 클럭 속도로 동작하는 것을 특징으로 한다(도 7 및 도 11 참조). 즉, 본 발명의 실시예에 따른 복호화 장치는 메모리 버퍼 46-10의 후단에 고속 메모리 버퍼 260을 추가적으로 구비시킴으로써, 터보 복호기의 1 클럭 동안에 상기 메모리 버퍼 46-10의 서로 다른 위치에 저장되어 있는 수신 데이터 비트들을 액세스하여 델타 메트릭 계산부 210의 각 계산기들 211~213으로 제공함으로써, 델타 메트릭 계산부 210에 의한 델타 메트릭 계산 동작이 상기 터보 복호기의 1 클럭 동안에 수행되도록 하는 것을 특징으로 한다.

<80>        이러한 본 발명의 실시예에 따른 복호화 장치는 1x EV-DV와 같이 고속의 패킷 데이터 서비스를 제공하는 통신시스템에 사용되는 경우에도 메모리 버퍼의 동작 주파수를 증가시킬 필요가 없도록 하는 것으로, 저전력 소모의 설계가 요구되는 이동통신 단말기의 구현에 적합하다 할 것이다. 참고적으로, 종래 기술의 경우 도 3에서와 같이 수신단의

메모리 버퍼 46-1에 연결된 디멀티플렉서 205를 통해 바로 터보 복호기를 연결하므로 도 6에서와 같이 수신단의 메모리 버퍼 46-1로부터 3번의 데이터 읽기 과정을 거치게 된다. 그러나, 본 발명에서 제안하고 있는 복호화 장치는 복호기의 입력단에 멀티플렉서 대신 메모리 버퍼 260을 추가적으로 구비시켜 한 클럭에 한번의 읽기 과정으로 SISO 복호기의 정상적인 동작이 가능하도록 구성한 것을 특징으로 한다. 즉, 본 발명의 복호화 장치는 3개의 델타 메트릭을 구하는데 한번의 메모리 읽기 동작만을 필요로 하는데, 이 동작은 터보 복호기의 동작 클럭과 동일하다. 이러한 동작은 고속 메모리 버퍼 260이 미리 수신신호를 입력받아 메모리에 저장하고 있다가 각 델타 메트릭 계산기들 211~213이 원하는 입력에 맞도록 수신 신호를 재구성하여 출력해 주기 때문에 가능하다.

<81> 하기에, 종래 기술에 따른 SISO 복호기의 구성에 대해서는 이미 앞서서 설명한 바 있으므로, 이에 대한 구체적인 설명은 생략하고 본 발명과 관련있는 고속 메모리 버퍼 260의 구성 및 그에 따른 동작에 초점을 맞추어 설명하기로 한다.

<82> 본 발명의 실시예에 따른 복호화 장치의 고속 메모리 버퍼 260은 후술될 도 9, 도 12 및 도 17에 도시된 바와 같이 구성될 수 있다. 도 9에 도시된 실시예는 고속 메모리 버퍼 260이 하나의 단방향 시프트 레지스터와, 윈도우 수 N에 각각 대응하는 양방향 시프트 레지스터들로 구성된 예를 나타낸다. 도 12에 도시된 실시예는 고속 메모리 버퍼 260이 하나의 단방향 시프트 레지스터와, 윈도우 수 2에 대응하는 양방향 시프트 레지스터들로 구성된 예를 나타낸다. 이러한 도 9 및 도 12에 도시된 구성은 동일한 원리에 따른 것이고, 다만 윈도우 수에 있어서는 차이가 있다. 도 17에 도시된 실시예는 고속 메모리 버퍼 260이 하나의 단방향 시프트 레지스터와, 윈도우 수 2에 대응하는 양방향 시프트 레지스터들로 구성된 예를 나타낸다. 이 도 17에 도시된 구성은 도 9 및 도 12에

도시된 구성과는 다른 원리에 따른 것이다. 도 9 및 도 12에 도시된 구성은 메모리 버퍼 46-10으로부터의 데이터 비트들이 단방향 시프트 레지스터와, 양방향 시프트 레지스터들로 동시에 입력되는 구조를 가진다. 반면에, 도 17에 도시된 구성은 메모리 버퍼 46-10으로부터의 데이터 비트들이 양방향 시프트 레지스터들로 입력되고, 단방향 시프트 레지스터로는 양방향 시프트 레지스터들로부터 출력되는 데이터 비트들이 입력되는 구조를 가진다.

<83> 하기에 제1 실시예는 도 9에 도시된 구성을 나타내며, 제2 실시예는 도 12에 도시된 구성을 나타내며, 제3 실시예는 도 17에 도시된 구성을 나타낸다. 비록, 제3 실시예의 확장 예에 대한 설명 및 구성이 도시되어 있지는 않지만, 동일한 원리에 따른 확장 예가 제1 실시예와 같이 가능할 것이다.

<84> 실시 예 1

<85> 도 9는 도 8에 도시된 고속 메모리 버퍼 260의 구성의 일 예를 보여주는 도면이다.

<86> 상기 도 9를 참조하면, 상기 고속 메모리 버퍼 260은 하나의 단방향 시프트 레지스터 310과, N개의 양방향 시프트 레지스터들 321~323과, 제어로직 330과, 역다중화기(DEMUX: Demultiplexer)들 341~343과, 다중화기(MUX: Multiplexer)들 351~353으로 구성된다.

<87>        상기 시프트 레지스터 310은 2NW의 저장 영역(길이)을 가지며, 데이터 입력을 위한 입력 단자와 데이터 출력을 위한 출력 단자를 구비한다. 여기서, N은 윈도우의 수이고, W는 윈도우의 크기이다. 상기 윈도우의 크기 W의 값은 크기 변화가 가능하다. 단 상기 윈도우의 크기 W가 변화하더라도 시프트 레지스터의 비(예: 2W, 4W)는 변화하지 않는다. W=24인 경우,  $\alpha$ 를 위한 시프트 레지스터 310은 96의 크기를 갖고,  $\beta_1$ 과  $\beta_2$ 를 위한 시프트 레지스터들 321~323은 48의 크기를 갖는다. W가 변경되면 그에 따라 시프트 레지스터의 길이가 변경되며, 데이터 처리의 흐름도 변경될 것이다. 그러나, W가 변한다고 하더라도 시프트 레지스터의 구조 형태는 변화하지 않는다. 상기 시프트 레지스터 310은 미리 설정된 클럭인 터보 복호기의 클럭에 따라 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 입력 단자를 통해 순차적으로 입력하고 좌측(side A)에서 우측(side B) 방향으로 시프트시킨다. 제1 길이(2NW)의 데이터 비트 열들이 형성되면, 상기 시프트 레지스터 310은 상기 형성된 제1 길이의 데이터 비트 열들을 상기 출력 단자를 통해 순차적으로 출력한다. 이 시프트 레지스터 310으로부터 출력되는 데이터 비트들은 알파 메트릭 계산부 220의 전단에 접속된 델타 메트릭 계산기 211로 인가된다.

<88>        상기 시프트 레지스터들 321~323 각각은 NW의 저장 영역을 가지며, 데이터 입/출력을 위한 제1 단자와 제2 단자를 구비한다. 여기서, 제1 단자는 상기 시프트 레지스터들 321~323 각각의 좌측에 구비된 단자를 나타내고, 제2 단자는 상기 시프트 레지스터들 321~323 각각의 우측에 구비된 단자를 나타내고, 상기 제1 단자 및 제2 단자는 데이터의 입력 뿐만 아니라 출력도 가능하다. 이러한 시프트 레지스터들 321~323의 수는 윈도우의 수에 의해 결정된다. 윈도우의 수가 N인 경우 상기 시프트 레지스터들 321~323의 수는 N으로 결정되고, 윈도우의 수가 2인 경우 상기 시프트 레지스터들 321~323의

수는 2로 결정된다. 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들은 상기 제1 길이의 1/2인 제2 길이(NW)의 비트들로 이루어지는 그룹들로 분할된다. 상기 시프트 레지스터들 321~323 각각은 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 좌측(side A)에서 우측(side B) 방향으로 순차적으로 입력하여 시프트시키고, 상기 제2 길이의 비트 열들이 형성되면 입력 순서와 반대의 순서인 우측에서 좌측 방향으로 상기 제1 단자를 통해 순차적으로 출력한다. 상기 시프트 레지스터들 321~323 각각은 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 우측에서 좌측 방향으로 순차적으로 입력하여 시프트시키고, 상기 제2 길이의 비트 열들이 형성되면 입력 순서와 반대의 순서인 좌측에서 우측 방향으로 상기 제2 단자를 통해 순차적으로 출력한다.

<89>        상기 메모리 버퍼 46-10의 출력단과 상기 시프트 레지스터들 321~323 각각의 사이에는 역다중화기들 341~343이 구비되고, 상기 시프트 레지스터들 321~323과 델타 메트릭 계산부 210의 사이에는 다중화기들 351~353이 구비된다. 상기 역다중화기들 341~343 각각의 입력단은 상기 메모리 버퍼 46-10의 출력단에 접속되고, 상기 역다중화기들 341~343 각각의 제1 출력 단자는 상기 각 시프트 레지스터들 321~323의 제2 단자에 접속되고, 상기 역다중화기들 341~343 각각의 제2 출력 단자는 상기 각 시프트 레지스터들 321~323의 제1 단자에 접속된다. 상기 다중화기들 351~353 각각의 제1 입력 단자는 상기 각 시프트 레지스터들 321~323의 제2 단자에 접속되고, 상기 다중화기들 351~353 각각의 제2 입력 단자는 상기 각 시프트 레지스터들 321~323의 제1 단자에 접속되고, 상기 다중화기들 351~353 각각의 출력 단자는 상기 델타 메트릭 계산기들 211~213에 접속된다.

<90>       상기 제어로직(control logic) 330은 상기 각 시프트 레지스터들 321~323과, 상기 역다중화기들 341~343과, 상기 다중화기들 351~353의 동작들을 제어하기 위한 선택 신호들 select1 ~ selectN을 제공한다. 이때 선택 신호들 각각은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 시프트 레지스터들 321~323 각각에 서로 다른 시점에 인가되도록 제어하기 위한 신호들로서 설정될 수 있다. 상기 제어로직 330은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 각 시프트 레지스터들 321~323의 제1 단자 또는 제2 단자로 입력되도록 한다.

<91>       예를 들어, 상기 제어로직 330은 입력 데이터 비트들이 홀수번째 그룹들의 비트들인 경우에는 "0" 또는 "logic low"레벨의 선택신호를 출력하고, 짝수번째 그룹들의 비트들인 경우에는 "1" 또는 "logic high"레벨의 선택신호를 출력한다. "0" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 341~343 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 321~323 각각의 제1 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 321~323은 제1 단자를 통해 입력되는 데이터 비트들을 순차적으로 좌측에서 우측 방향(오른쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 321~323은 이전에 제2 단자를 통해 입력된 후 저장되어 있는 NW개의 데이터 비트들을 다시 좌측에서 우측 방향으로 순차적으로 시프트시켜 상기 제2 단자를 통해 출력되도록 한다.

<92>       "1" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 341~343 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 321~323 각

각의 제2 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 321~323은 제2 단자를 통해 입력되는 데이터 비트들을 순차적으로 우측에서 좌측 방향(왼쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 321~323은 이전에 제1 단자를 통해 입력된 후 저장되어 있는 NW개의 데이터 비트들을 다시 우측에서 좌측 방향으로 순차적으로 시프트시켜 상기 제1 단자를 통해 출력되도록 한다.

<93>       상기 각 시프트 레지스터들 321~323의 제1 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 351~353의 제2 입력 단자로 인가되고, 제2 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 351~353의 제1 입력 단자로 인가된다. 상기 각 다중화기 351~353은 제1 입력 단자 및 제2 입력 단자를 통해 인가되는 데이터 비트들을 다중화하여 해당하는 각 델타 매트릭 계산기들 211~213으로 출력한다.

<94>       전술한 바와 같이, 본 발명의 실시예에 따른 복호화 장치는 메모리 버퍼 46-10으로부터 서로 다른 위치에 저장되어 있는 데이터 비트들을 도 9에 도시된 바와 같은 구성을 가지는 고속 메모리 버퍼 260을 사용하여 SISO 복호기로 입력되도록 한다. 즉, 상기 고속 메모리 버퍼 260은 미리 순차적으로 입력된 데이터 비트들의 순서를 시프트 레지스터들 310, 321~323을 통해서 윈도우 모드의 SISO 복호기가 요구하는 순서대로 재구성한다.

<95>       다시 도 9를 참조하면, 메모리 버퍼 46-10으로부터의 M 비트가 2NW 또는 NW의 저장 영역을 가지는 시프트 레지스터들 310, 321~323에 입력된다. 여기서, N은 윈도우의 수를 나타내고, W는 윈도우의 크기를 나타내고, M은 메모리 버퍼 46-10으로부터 터보 복호기의 1클럭 동안에 입력되는 데이터 비트들의 비트 수를 나타낸다. 상기 메모리 버퍼 46-10이 QCTC 메모리 버퍼인 경우, M은 시스템 부호(systematic code)와 패리티 부호들



(parity codes)의 비트 폭(bit width)을 모두 합한 값이다. 즉, M비트의 신호는 시스템 부호(systematic code) M/3 비트, 제1 패리티 부호(parity1 code) M/3 비트, 제2 패리티 부호(parity2 code) M/3 비트를 모두 합하여 구성되는 신호이다. M비트 신호의 입력에 대하여 모든 시프트 레지스터들 310, 321~323, 역다중화기들 341~343, 다중화기들 351~353은 M비트의 폭을 가지는 것으로 구성된다. 이 M비트 신호의 출력, 즉 시프트 레지스터들 310, 321~323에서 출력되는 값은 델타 메트릭 계산기들 211~213의 입력으로 제공된다. 상기 각 델타 메트릭 계산기들 211~213은 모두 합쳐 M 비트인 3개의 M/3 비트의 신호를 입력한다.

<96> 도 10은 도 9에 도시된 제어로직 330에 의한 제어 동작을 보여주는 도면이다. 이 도면은 구체적으로는 상기 제어로직 330이  $\beta$  메트릭을 위한 양방향 시프트 레지스터들 321~323과, 이 레지스터들 321~323의 전단 및 후단에 접속된 역다중화기들 341~343과, 다중화기들 351~353을 제어하는 흐름을 보여주는 도면이다.

<97> 상기 도 10에서, 1011단계~1017단계의 동작은 시프트 레지스터 321과, 역다중화기 341과, 다중화기 351을 제어하는 동작의 처리 흐름을 나타낸다. 1021단계~1027단계의 동작은 시프트 레지스터 322와, 역다중화기 342와, 다중화기 352를 제어하는 동작의 처리 흐름을 나타낸다. 1031단계~1037단계의 동작은 시프트 레지스터 323과, 역다중화기 343과, 다중화기 353을 제어하는 동작의 처리 흐름을 나타낸다. 상기 각 처리 흐름은 시작되는 시점에서, 그리고 사용되는 신호들의 이름에 있어서만 차이가 있을 뿐 그 수행되는 동작에 있어서는 동일하므로, 하기에서는 1011단계~1017단계의 처리 흐름에 대해서만 설명하기로 한다. 이러한 제어 동작을 위해 상기 제어로직 330의 내부에는 각 시프트 레지스터들 321~323에 대응하는 카운터들이 구비된다. 이 카운터들은 미리 설정된 W만

크의 오프셋(offset)을 가지고 서로 다른 시점에서 초기화되어 카운트 동작을 수행한다. 시프트 레지스터 321에 대응하는 카운터(counter1)는  $T=0$ 인 시점에서 초기화되고, 시프트 레지스터 322에 대응하는 카운터(counter2)는  $T=Wt$ 인 시점에서 초기화되고, 시프트 레지스터 323에 대응하는 카운터(counterN)는  $T=(N-1)Wt$ 인 시점에서 초기화된다. 여기서,  $t$ 는 시간, 즉 단위 클럭을 나타낸다.

<98>        상기 도 10을 참조하면, 1011단계에서 제어로직 330은 시프트 레지스터 321을 초기화한다. 이 초기화 동작시 카운터(counter1)의 카운트값은  $count1=0$ 으로 초기화되고, 선택신호는  $select1=0$ 으로 초기화된다. 또한, 시프트 레지스터 321의 좌측(side A) 단자가 입력 단자로서 설정되고, 우측(side B) 단자가 출력 단자로서 설정된다. 1012단계에서 상기 제어로직 330은 도 8의 메모리 버퍼 46-10을 액세스하여 데이터 비트들을 읽어들이는 다. 1013단계에서 상기 제어로직 330은 카운트값  $count1=NW$ 인지 여부를 판단함으로써 시프트 레지스터 321이 가득 찼는지 여부를 판단한다. 카운트값  $count1$ 이  $NW$ 인 경우에, 상기 제어로직 330은 1014단계에서 카운트값  $count1$ 을 0으로 설정한다. 카운트값  $count1$ 이  $NW$ 가 아닌 경우에, 상기 제어로직 330은 1015단계에서 카운트값  $count1$ 을 1 증가시킨다. 상기 1014단계를 수행한 이후에, 1016단계에서 상기 제어로직 330은  $select1$  신호를 반전시킨다. 즉,  $select1$ 신호를 비트 반전하여, "1"의  $select1$  신호는 "0"의  $select1$  신호로 변환하고, "0"의  $select1$  신호는 "1"의  $select1$  신호로 변환한다. 상기 1016단계의 동작에 의해 데이터 비트들의 입출력 방향과 시프트 방향이 바뀌게 된다. 상기 1016단계를 수행한 이후 또는 상기 1015단계를 수행한 이후에, 1017단계에서 상기 제어로직 330은 상기 메모리 버퍼 46-10으로부터 입력되는 데이터 비트들을 상기 시프트 레지스터

321에 라이트한다. 1017단계를 수행한 이후에 상기 제어로직 330은 1012단계로 되돌아가 전술한 동작을 반복 수행한다.

<99> 도 11은 도 9에 도시된 고속 메모리 버퍼 260에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면이다.

<100> 상기 도 11을 참조하면, 메모리 버퍼 46-10의 서로 다른 주소에 저장되어 있는 데이터 비트들이 고속 메모리 버퍼 260에 의해 액세스된다. 여기서는 상기 고속 메모리 버퍼 260이 상기 메모리 버퍼 46-10의 3가지 주소들에 저장되어 있는 데이터 비트들 data1, data2, data3을 액세스하는 예를 나타내고 있다. 이와 같이 3가지의 데이터 비트들 data1, data2 data3 이 모두 터보 복호기의 1 클럭내의 동일 시점에서 상기 고속 메모리 버퍼 260에 들어오게 되면, 델타 메트릭 계산부 210의 각 계산기들 211~213은 동일한 시점에서 델타 메트릭을 계산하는 동작을 수행한다. 데이터 비트 data1(M bit)은 systematic code(M/3 bit) + parity1 code(M/3 bit) + parity2 code(M/3 bit)을 의미하며, data2, data3도 동일하다.

<101> 실시 예 2

<102> 도 12는 도 8에 도시된 고속 메모리 버퍼 260의 구성의 다른 예를 보여주는 도면이다. 이 도면은 윈도우의 수가 N=2일 때, 즉 베타가 2개의 윈도우를 가질 때 고속 메모리 버퍼 260의 구성을 보여주는 도면이다.

- <103>      상기 도 12를 참조하면, 상기 고속 메모리 버퍼 260은 하나의 단방향 시프트 레지스터 410과,  $N=2$ 개의 양방향 시프트 레지스터들 421, 422와, 제어로직 430과, 역다중화기들 441, 442와, 다중화기 451, 452로 구성된다.
- <104>      상기 시프트 레지스터 410은  $2NW=4W$ 의 저장 영역(길이)을 가지며, 데이터 입력을 위한 입력 단자와 데이터 출력을 위한 출력 단자를 구비한다. 상기 시프트 레지스터 410은 터보 복호기의 클럭에 따라 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 입력 단자를 통해 순차적으로 입력하고 좌측(side A)에서 우측(side B) 방향으로 시프트시킨다. 제1 길이( $4W$ )의 데이터 비트 열들이 형성되면, 상기 시프트 레지스터 410은 상기 형성된 제1 길이의 데이터 비트 열들을 상기 출력 단자를 통해 순차적으로 출력한다. 이 시프트 레지스터 410으로부터 출력되는 데이터 비트들은 알파 메트릭 계산부 220의 전단에 접속된 델타 메트릭 계산기 211로 인가된다.
- <105>      상기 시프트 레지스터들 421, 422 각각은  $NW=2W$ 의 저장 영역을 가지며, 데이터 입/출력을 위한 제1 단자와 제2 단자를 구비한다. 여기서, 제1 단자는 상기 시프트 레지스터들 421, 422 각각의 좌측에 구비된 단자를 나타내고, 제2 단자는 상기 시프트 레지스터들 421, 422 각각의 우측에 구비된 단자를 나타내고, 상기 제1 단자 및 제2 단자는 데이터의 입력 뿐만 아니라 출력도 가능하다. 이러한 시프트 레지스터들 421, 422의 수는 윈도우의 수에 의해 결정된다. 윈도우의 수가  $N=2$ 인 경우 상기 시프트 레지스터들 421, 422의 수는 2로 결정된 것이다. 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들은 상기 제1 길이의  $1/2$ 인 제2 길이( $2W$ )의 비트들로 이루어지는 그룹들로 분할된다. 상기 시프트 레지스터들 421, 422 각각은 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 좌측(side A)에서 우측(side B) 방향으로 순차적으로 입력하여 시

프트시키고, 상기 제2 길이의 비트 열들이 형성되면 입력 순서와 반대의 순서인 우측에서 좌측 방향으로 상기 제1 단자를 통해 순차적으로 출력한다. 상기 시프트 레지스터들 421,422 각각은 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 우측에서 좌측 방향으로 순차적으로 입력하여 시프트시키고, 상기 제2 길이의 비트 열들이 형성되면 입력 순서와 반대의 순서인 좌측에서 우측 방향으로 상기 제2 단자를 통해 순차적으로 출력한다.

<106>       상기 메모리 버퍼 46-10의 출력단과 상기 시프트 레지스터들 421,422 각각의 사이에는 역다중화기들 441,442가 구비되고, 상기 시프트 레지스터들 421,422와 델타 메트릭 계산부 210의 사이에는 다중화기들 451,452가 구비된다. 상기 역다중화기들 441,442 각각의 입력단은 상기 메모리 버퍼 46-10의 출력단에 접속되고, 상기 역다중화기들 441,442 각각의 제1 출력 단자는 상기 각 시프트 레지스터들 421,422의 제2 단자에 접속되고, 상기 역다중화기들 451,452 각각의 제2 출력 단자는 상기 각 시프트 레지스터들 421,422의 제1 단자에 접속된다. 상기 다중화기들 451,452 각각의 제1 입력 단자는 상기 각 시프트 레지스터들 421,422의 제2 단자에 접속되고, 상기 다중화기들 451,452 각각의 제2 입력 단자는 상기 각 시프트 레지스터들 421,422의 제1 단자에 접속되고, 상기 다중화기들 451,452 각각의 출력 단자는 상기 델타 메트릭 계산기들 211~213에 접속된다.

<107>       상기 제어로직 430은 상기 각 시프트 레지스터들 421,422와, 상기 역다중화기들 441,442와, 상기 다중화기들 451,452의 동작들을 제어하기 위한 선택 신호들 `select1`, `select2`를 제공한다. 이때 선택 신호들 각각은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 시프트 레지스터들 421,422 각각에 서로 다른 시점에 인가되도록

제어하기 위한 신호들로서 설정될 수 있다. 상기 제어로직 430은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 각 시프트 레지스터들 421,422의 제1 단자 또는 제2 단자로 입력되도록 한다.

<108> 예를 들어, 상기 제어로직 430은 입력 데이터 비트들이 홀수번째 그룹들의 비트들인 경우에는 "0" 또는 "logic low"레벨의 선택신호를 출력하고, 짝수번째 그룹들의 비트들인 경우에는 "1" 또는 "logic high"레벨의 선택신호를 출력한다. "0" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 441,442 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 421,422 각각의 제1 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 421,422는 제1 단자를 통해 입력되는 데이터 비트들을 순차적으로 좌측에서 우측 방향(오른쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 421,422는 이전에 제2 단자를 통해 입력된 후 저장되어 있는 2W개의 데이터 비트들을 다시 좌측에서 우측 방향으로 순차적으로 시프트시켜 상기 제2 단자를 통해 출력되도록 한다.

<109> "1" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 441,442 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 421,422 각각의 제2 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 421,422는 제2 단자를 통해 입력되는 데이터 비트들을 순차적으로 우측에서 좌측 방향(왼쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 421,422는 이전에 제1 단자를 통해

입력된 후 저장되어 있는 2W개의 데이터 비트들을 다시 우측에서 좌측 방향으로 순차적으로 시프트시켜 상기 제1 단자를 통해 출력되도록 한다.

<110>        상기 각 시프트 레지스터들 421,422의 제1 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 451,452의 제2 입력 단자로 인가되고, 제2 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 451,452의 제1 입력 단자로 인가된다. 상기 각 다중화기 451,452는 제1 입력 단자 및 제2 입력 단자를 통해 인가되는 데이터 비트들을 다중화하여 해당하는 각 델타 메트릭 계산기들 211~213으로 출력한다.

<111>        다시 도 8을 참조하면, 제1 메트릭 계산부 210의 각 계산기들 211~213은 상기 각 시프트 레지스터들 410,421,422로부터의 출력 비트들을 입력하고 해당하는 델타 메트릭들을 각각 계산한다. 제2 메트릭 계산부 220은 상기 시프트 레지스터 410에 대응하는 상기 제1 메트릭 계산부 210의 계산기 211로부터의 델타 메트릭을 입력하여 알파 메트릭을 계산한다. 제3 메트릭 계산부 230은 상기 시프트 레지스터들 421,422 각각에 대응하는 상기 제1 메트릭 계산부 210의 계산기들 212,213으로부터의 델타 메트릭들을 입력하여 베타 메트릭들을 각각 계산한다. 제4 메트릭 계산부 240의 각 계산기들 241~243은 상기 알파 메트릭을 입력하고, 또한 다중화기 233에 의한 상기 베타 메트릭들의 다중화 결과를 입력하여, 상기 각 시프트 레지스터들 410,421,422에 대응하는 LLR 값들을 계산한다. 감산부 250의 각 감산기들 251~253은 상기 각 LLR 값으로부터 상기 시프트 레지스터 410의 출력을 감산하고, 감산 결과들을 인터리빙/디인터리빙을 위해 출력한다.

<112>        도 13은 도 12에 도시된 제어로직 430에 의한 제어 동작을 보여주는 도면이다. 이 도면은 구체적으로는 상기 제어로직 430이  $\beta$  메트릭을 위한 양방향 시프트 레지스터들

421,422와, 이 레지스터들 421,422의 전단 및 후단에 접속된 역다중화기들 441,422와, 다중화기들 451,452를 제어하는 흐름을 보여주는 도면이다.

<113>       상기 도 13에서, 1111단계~1117단계의 동작은 시프트 레지스터 421과, 역다중화기 441과, 다중화기 451을 제어하는 동작의 처리 흐름을 나타낸다. 1121단계~1127단계의 동작은 시프트 레지스터 422와, 역다중화기 442와, 다중화기 452를 제어하는 동작의 처리 흐름을 나타낸다. 상기 각 처리 흐름은 시작되는 시점에서, 그리고 사용되는 신호들의 이름에 있어서만 차이가 있을 뿐 그 수행되는 동작에 있어서는 동일하므로, 하기에서는 1111단계~1117단계의 처리 흐름에 대해서만 설명하기로 한다. 이러한 제어 동작을 위해 상기 제어로직 430의 내부에는 각 시프트 레지스터들 421,422에 대응하는 카운터들이 구비된다. 이 카운터들은 미리 설정된 W만큼의 오프셋(offset)을 가지고 서로 다른 시점에서 초기화되어 카운트 동작을 수행한다. 시프트 레지스터 421에 대응하는 카운터(counter1)는  $T=0$ 인 시점에서 초기화되고, 시프트 레지스터 422에 대응하는 카운터(counter2)는  $T=Wt$ 인 시점에서 초기화된다. 여기서,  $t$ 는 시간, 즉 단위 클럭을 나타낸다.

<114>       상기 도 13을 참조하면, 1111단계에서 제어로직 430은 시프트 레지스터 421을 초기화한다. 이 초기화 동작시 카운터(counter1)의 카운트값은  $count1=0$ 으로 초기화되고, 선택신호는  $select1=0$ 으로 초기화된다. 또한, 시프트 레지스터 421의 좌측(side A) 단자가 입력 단자로서 설정되고, 우측(side B) 단자가 출력 단자로서 설정된다. 1112단계에서 상기 제어로직 430은 도 8의 메모리 버퍼 46-10을 액세스하여 데이터 비트들을 읽어들이는 다. 1113단계에서 상기 제어로직 430은 카운트값  $count1=2W$ 인지 여부를 판단함으로써 시프트 레지스터 421이 가득 찼는지 여부를 판단한다. 카운트값  $count1$ 이  $2W$ 인 경우에, 상



기 제어로직 430은 1114단계에서 카운트값 count1을 0으로 설정한다. 카운트값 count1이 2W가 아닌 경우에, 상기 제어로직 430은 1115단계에서 카운트값 count1을 1 증가시킨다. 상기 1114단계를 수행한 이후에, 1116단계에서 상기 제어로직 430은 select1 신호를 반전시킨다. 즉, select1신호를 비트 반전하여, "1"의 select1 신호는 "0"의 select1 신호로 변환하고, "0"의 select1 신호는 "1"의 select1 신호로 변환한다. 상기 1116단계의 동작에 의해 데이터 비트들의 입출력 방향과 시프트 방향이 바뀌게 된다. 상기 1116단계를 수행한 이후 또는 상기 1115단계를 수행한 이후에, 1117단계에서 상기 제어로직 430은 상기 메모리 버퍼 46-10으로부터 입력되는 데이터 비트들을 상기 시프트 레지스터 421에 라이트한다. 1117단계를 수행한 이후에 상기 제어로직 430은 1112단계로 되돌아가 전술한 동작을 반복 수행한다.

<115> 도 14는 도 12에 도시된 알파 메트릭을 위한 시프트 레지스터 410에 의한 데이터 처리 흐름을 보여주는 도면이다.

<116> 상기 도 14를 참조하면, 상기 시프트 레지스터 410은 도 8의 메모리 버퍼 46-10으로부터의 데이터 비트들을 좌측에서 우측 방향으로 순차적으로 입력 및 시프트시킨다. "side A"는 데이터 비트들이 입력되는 위치를 나타내고, "side B"는 데이터 비트들이 출력되는 위치를 나타낸다. 상기 시프트 레지스터 410으로 데이터 비트들이 입력되면, 이 입력 데이터 비트들은 4W만큼 지연된 후 출력된다. 이 시프트 레지스터 410은 단순히 FIFO(First-In First-Out)의 구조를 갖는다.

<117> 도 15는 도 12에 도시된 베타 메트릭을 위한 시프트 레지스터 421에 대한 데이터 처리 흐름을 보여주는 도면이다. 이 처리 흐름은 베타 메트릭을 위한 다른 시프트 레지스터 422에 대한 데이터 처리 흐름과 동일하다.

<118>       상기 도 15를 참조하면, 상기 시프트 레지스터 421은 메모리 버퍼 46-10으로부터 순차적으로 읽어들이는 데이터를 터보 복호기가 필요로 하는 시점을 맞추기 위해서 미리 설정된 시간동안 지연시킨 후 출력한다. 상기 시프트 레지스터 421은 순차적으로 입력되는 데이터 비트들을 저장한다. 상기 시프트 레지스터 421이 가득 차게 되면, 이전에 데이터 비트들이 입력된 방향과 반대 방향으로 저장되어 있는 데이터 비트들을 시프트시키면서 출력하여 그 출력 데이터 비트들이 해당하는 델타 메트릭 계산기 212로 제공되도록 한다. 이에 따라 상기 시프트 레지스터 421의 반대쪽에는 빈 공간이 발생하게 된다. 이러한 빈 공간은 이전에 데이터 비트들이 입력된 방향과 반대 방향으로 새로운 데이터 비트들이 입력되어 다시 가득 채워지게 된다. 이와 같이 상기 시프트 레지스터 421의 좌우로 왕복하면서 데이터가 입력 및 출력의 동작이 반복되고, 결과적으로 도면에 나타낸 바와 같은 데이터의 흐름에 따라 데이터 비트들이 출력되어 해당하는 델타 메트릭 계산기 212로 제공된다.

<119>       도 16은 도 12에 도시된 고속 메모리 버퍼 260에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면이다.

<120>       상기 도 16에서, "delta block input for alpha"는 도 12의 단방향 시프트 레지스터 410의 입/출력 데이터 비트들을 나타내는 것으로, side A in은 입력 데이터 비트들을 나타내고, side B out은 출력 데이터 비트들을 나타낸다. "delta block input for beta1"은 양방향 시프트 레지스터 421의 입/출력 데이터 비트들을 나타내고, "delta block input for beta2"는 양방향 시프트 레지스터 422의 입/출력 데이터 비트들을 나타낸다. 상기 "delta block input for beta1" 및 "delta block input for beta1"에서, "side A in" 및 "side A out"는 제1 단자를 통해 입/출력되는 데이터 비트들을 나타내고,

"side B in" 및 "side B out"는 제2 단자를 통해 입/출력되는 데이터 비트들을 나타낸다. "select1" 및 "select2"는 제어 로직 430에 의해 생성된 후 상기 시프트 레지스터들 421, 422 각각에 제공되는 제어신호들을 나타낸다. "a out", " $\beta 1$  out", " $\beta 2$  out"은 최종 출력되는 메트릭들을 나타내는 것으로, 이러한 출력 메트릭들을 이용하여 LLR 값이 구해진다.

<121> "delta block input for beta1" 부분의 경우, 초기에는 side A를 통해서 데이터 비트들이 입력된다. 데이터 비트들이 a부터 차례대로 b, c, d, ..., h 의 순서로 입력되고, 초기 구간부터 2W만큼 경과한 후에 side A에서 데이터 비트들이 출력된다. side A에서 출력 동작이 수행됨과 동시에, side B를 통해서도 입력 동작이 수행된다. 이것은 시프트 레지스터 421이 시프트 방향을 바꿔서 동작하고 있다는 것을 의미한다. 다시 2W만큼 경과한 후에 side A에서 다시 입력 동작이 수행되어 q번째 데이터 비트가 입력될 때, side B에는 p번째 데이터 비트가 출력된다.

<122> "delta block input for beta2" 부분에서의 동작은 상기 "delta block input for beta1" 부분에서의 동작과 동일한 방식으로 이루어진다. 다만, 시프트 레지스터 422가 초기화되는 시점이 시프트 레지스터 421과는 차이가 있으므로, 데이터 비트들의 입/출력이 동일한 구간에서 수행되지는 않는다.

<123> "delta block input for alpha"의 경우, side A에서는 데이터 비트들의 입력 동작만이 수행되고, side B에서는 데이터 비트들의 출력 동작만이 수행된다. 초기 구간부터 4W가 경과한 시점에서 최초로 입력되었던 a번째 데이터 비트가 출력된다.

<124> 각 시프트 레지스터들 410, 421, 422의 출력 데이터 비트들을 도 6에 도시된 출력 데이터 비트들과 비교해 보면 동일한 흐름으로 출력되는 것을 알 수 있다. 다만, 도 16의

출력 방식이 도 6에 도시된 출력 방식에 비해서 2W만큼의 초기 지연을 발생한다는 차이점이 있다. 그러나, 이러한 차이점은 고속 메모리 버퍼 260을 초기에 동작시키는 동안에 발생하는 것으로, 터보 복호기의 복호화가 시작되는 초기에만 발생하므로 복호화 성능에는 별다른 영향이 없다.

<125>      실시 예 3

<126>      도 17은 도 8에 도시된 고속 메모리 버퍼 260의 구성의 또 다른 예를 보여주는 도면이다.

<127>      상기 도 17을 참조하면, 상기 고속 메모리 버퍼 260은 제1 스테이지의 양방향 시프트 레지스터들 521,522와, 제2 스테이지의 양방향 시프트 레지스터 510과, 제어로직 530과, 역다중화기들 541,542와, 다중화기들 551~553으로 구성된다.

<128>      상기 제1 스테이지의 양방향 시프트 레지스터들 521,522는 각각 NW의 저장영역(길이)를 가지며, 데이터 입/출력을 위한 제1 단자 및 제2 단자를 구비한다. 여기서, N은 윈도우의 수를 나타내고, W는 윈도우의 크기를 나타낸다. 상기 시프트 레지스터들의 수는 윈도우의 수에 의해 결정되고, 상기 저장영역은 윈도우의 수와 윈도우의 크기의 곱에 의해 결정된다. 상기 제1 단자는 상기 각 시프트 레지스터들 521,522 각각의 좌측에 구비된 단자, 즉 side A의 단자를 나타내고, 상기 제2 단자는 상기 각 시프트 레지스터들 521,522 각각의 우측에 구비된 단자, 즉 side B의 단자를 나타낸다. 상기 제1 단자 및 제2 단자는 데이터의 입력 뿐만 아니라 출력도 가능하다. 도 8의 메모리 버퍼 46-10으로

부터의 입력 데이터 비트들은 상기 길이의 비트들로 이루어지는 그룹들로 분할된다. 상기 시프트 레지스터들 521, 522는 각각 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 좌측(side A)에서 우측(side B) 방향으로 순차적으로 입력하여 시프트시키고, 상기 길이의 비트 열들이 형성되면 이 형성된 비트 열들을 입력 순서와 반대의 순서인 우측에서 좌측 방향으로 시프트시켜 상기 제1 단자를 통해 순차적으로 출력한다. 상기 시프트 레지스터들 521, 522는 각각 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 우측(side B)에서 좌측(side A) 방향으로 순차적으로 입력하여 시프트시키고, 상기 길이의 비트 열들이 형성되면 이 형성된 비트 열들을 입력 순서와 반대의 순서인 좌측에서 우측 방향으로 시프트시켜 상기 제2 단자를 통해 순차적으로 출력한다.

<129>        상기 제2 스테이지의 양방향 시프트 레지스터 510은 NW의 저장 영역을 가지며, 데이터 입/출력을 위한 제3 단자 및 제4 단자를 구비한다. 상기 시프트 레지스터 510은 상기 제1 단자를 통해 순차적으로 출력되는 비트들을 상기 제3 단자를 통해 입력하여 좌측에서 우측 방향으로 순차적으로 시프트시킨다. 상기 길이의 비트 열들이 형성되면, 상기 시프트 레지스터 510은 입력 순서와 반대의 순서인 우측에서 좌측 방향으로 상기 형성된 비트 열들을 시프트시켜 상기 제3 단자를 통해 순차적으로 출력한다. 상기 시프트 레지스터 510은 상기 제2 단자를 통해 순차적으로 출력되는 비트들을 상기 제4 단자를 통해 입력하여 우측에서 좌측 방향으로 순차적으로 시프트시킨다. 상기 길이의 비트 열들이 형성되면, 상기 시프트 레지스터 510은 입력 순서와 반대의 순서인 좌측에서 우측 방향으로 상기 형성된 비트 열들을 시프트시켜 상기 제4 단자를 통해 순차적으로 출력한다.

<130>       상기 메모리 버퍼 46-10의 출력단과 상기 시프트 레지스터들 521,522 각각의 사이에는 역다중화기들 541,542가 구비되고, 상기 시프트 레지스터들 510,521,522와 델타 메트릭 계산부 210의 사이에는 다중화기들 551~553이 구비된다. 상기 역다중화기들 541,542 각각의 입력단은 상기 메모리 버퍼 46-10의 출력단에 접속되고, 상기 역다중화기들 541,542 각각의 제1 출력 단자는 상기 각 시프트 레지스터들 521,522의 제2 단자에 접속되고, 상기 역다중화기들 541,542 각각의 제2 출력 단자는 상기 각 시프트 레지스터들 521,522의 제1 단자에 접속된다. 상기 다중화기들 551~553 각각의 제1 입력 단자는 상기 각 시프트 레지스터들 510,521,522의 제2 단자에 접속되고, 상기 다중화기들 551~553 각각의 제2 입력 단자는 상기 각 시프트 레지스터들 510,521,522의 제1 단자에 접속되고, 상기 다중화기들 551~553 각각의 출력 단자는 상기 델타 메트릭 계산기들 211~213에 접속된다.

<131>       상기 제어로직 530은 상기 각 시프트 레지스터들 510,521,522와, 상기 역다중화기들 541,542와, 상기 다중화기들 551~553의 동작들을 제어하기 위한 선택 신호들 select1, select2, select3을 제공한다. 이때 선택 신호들 각각은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 시프트 레지스터들 521,522 각각에 서로 다른 시점에 인가되도록 제어하기 위한 신호들로서 설정될 수 있다. 상기 제어로직 530은 상기 메모리 버퍼 46-10으로부터의 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단한다. 그 판단 결과에 따라 상기 제어 로직 530은 상기 입력 데이터 비트들을 상기 제1 단자 또는 상기 제2 단자로 인가되도록 제어하기 위한 선택신호들 select2,select3을 상기 시프트 레지스터들 521,522로 각각 제공한다.

<132> 예를 들어, 상기 제어로직 530은 입력 데이터 비트들이 홀수번째 그룹들의 비트들인 경우에는 "0" 또는 "logic low"레벨의 선택신호를 출력하고, 짝수번째 그룹들의 비트들인 경우에는 "1" 또는 "logic high"레벨의 선택신호를 출력한다. "0" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 541,542 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 521,522 각각의 제1 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 521,522는 제1 단자를 통해 입력되는 데이터 비트들을 순차적으로 좌측에서 우측 방향(오른쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 521,522는 이전에 제2 단자를 통해 입력된 후 저장되어 있는 2W개의 데이터 비트들을 다시 좌측에서 우측 방향으로 순차적으로 시프트시켜 상기 제2 단자를 통해 출력되도록 한다.

<133> "1" 레벨의 선택신호가 출력되는 경우, 상기 각 다중화기들 541,542 각각은 메모리 버퍼 46-10으로부터의 입력 데이터 비트들을 상기 각 시프트 레지스터들 521,522 각각의 제2 단자로 입력되도록 한다. 이에 따라 상기 각 시프트 레지스터들 521,522는 제2 단자를 통해 입력되는 데이터 비트들을 순차적으로 우측에서 좌측 방향(왼쪽 방향)으로 시프트시킨다. 이와 동시에, 상기 각 시프트 레지스터들 521,522는 이전에 제1 단자를 통해 입력된 후 저장되어 있는 2W개의 데이터 비트들을 다시 우측에서 좌측 방향으로 순차적으로 시프트시켜 상기 제1 단자를 통해 출력되도록 한다.

<134> 상기 각 시프트 레지스터들 521,522의 제1 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 552,553의 제2 입력 단자로 인가되고, 제2 단자를 통해 출력되는 데이터 비트들은 상기 각 다중화기 552,553의 제1 입력 단자로 인가된다. 상기 각 다중화기

552,553은 제1 입력 단자 및 제2 입력 단자를 통해 인가되는 데이터 비트들을 다중화하여 해당하는 각 델타 메트릭 계산기들 211~213으로 출력한다.

<135>       상기 시프트 레지스터 521의 제1 단자를 통해 출력되는 데이터 비트들은 상기 시프트 레지스터 510의 제1 단자로 또한 입력되고, 제2 단자를 통해 출력되는 데이터 비트들은 상기 시프트 레지스터 510의 제2 단자로 또한 입력된다. 상기 시프트 레지스터 510의 동작은 상기 시프트 레지스터들 521,522의 동작과 동일하다. 상기 시프트 레지스터 510의 제1 단자를 통해 출력되는 데이터 비트들은 상기 다중화기 551의 제2 입력 단자로 인가되고, 제2 단자를 통해 출력되는 데이터 비트들은 상기 다중화기 551의 제1 입력 단자로 인가된다. 상기 다중화기 551로부터 출력되는 데이터 비트들은 알파 메트릭 계산부 220의 전단에 접속된 델타 메트릭 계산기 211로 인가된다.

<136>       전술한 실시예는 도 12에 도시된 고속 메모리 버퍼 260과 구성을 달리할 뿐 동작은 동일하게 수행된다. 이 실시예에 따르면,  $\alpha$ 를 위한 시프트 레지스터 510이 도 9 및 도 12에 도시된 실시예와 달리 2W의 길이를 가지는 양방향 시프트 레지스터이고, 이 시프트 레지스터 510이 메모리 버퍼 46-10으로부터의 데이터 비트들을 제공받는 것이 아니라,  $\beta_1$ 을 위한 시프트 레지스터 521로부터의 데이터 비트들을 제공받아 다중화기 551을 통해 출력한다는 차이점이 있다. 즉,  $\beta_1$  메트릭을 위한 시프트 레지스터 521의 출력을 동일한 구조의 시프트 레지스터 510이 다시 입력받는 형태를 갖는다.  $\beta_1$  메트릭을 위한 시프트 레지스터 521이 실제 입력되는 신호의 역순으로 배치된다면 그것을 다시 역순으로 재구성하여, 원래 입력 신호의 순서로 원상 복귀시켜주어  $\alpha$  메트릭을 위한 시프트 레지스터 510에 입력될 수 있도록 한 것이다.  $\alpha$  메트릭을 위한 시프트 레지스터 510과,



$\beta 1$  메트릭을 위한 시프트 레지스터 521이 항상 반대 방향으로 동작하도록 구성하면, 도 12에 도시된 방식에 비해  $2W$ 만큼의 시프트 레지스터의 크기를 줄일 수 있다.

<137> 도 18은 도 17에 도시된 제어로직 530에 의한 제어 동작을 보여주는 도면이다. 이 도면에 도시된 제어 동작의 처리 흐름은 도 13에 도시된 처리 흐름과 유사하다. 다만, 차이가 있다면, 알파를 위한 시프트 레지스터 510을 제어하는 흐름이 추가되었다는 것이며, 각 제어 흐름에서는 초기화 과정을 나타내는 1211단계, 1221단계 및 1231단계에서 차이점이 있음을 알 수 있다.

<138> 상기 도 18을 참조하면, 알파를 위한 시프트 레지스터 510은  $T=(2W-1)t$  시점에서 초기화되고, 베타(beta1)를 위한 시프트 레지스터 521은  $T=0$  시점에서 초기화되고, 베타(beta2)를 위한 시프트 레지스터 522는  $T=(W-1)t$  시점에서 초기화된다. 즉, 상기 시프트 레지스터 510은  $(2W-1)$  클럭 후에 초기화되고, 상기 시프트 레지스터 521은 0 클럭에서 초기화되고, 상기 시프트 레지스터 522는  $(W-1)$  클럭 후에 초기화된다. 시프트 레지스터들의 전체 동작주기가  $4W$ 인 것을 감안하면 두 개의 시프트 레지스터들  $2W$  주기로 반대 방향으로 동작하는 것을 알 수 있다. 이러한 초기화 과정을 제외하고는 도 13에 도시된 동작과 동일하므로 이에 대한 구체적인 설명을 생략하기로 한다.

<139> 도 19는 도 17에 도시된 시프트 레지스터 510에 의한 데이터 처리 흐름을 보여주는 도면이다.

<140> 상기 도 19를 참조하면, 상기 시프트 레지스터 510에 의한 데이터 처리 동작은 도 15에 도시된 바와 같이 베타 메트릭을 위한 시프트 레지스터들에 의한 데이터 처리 동작과 동일하게 수행됨을 알 수 있다.

- <141> 도 20은 도 17에 도시된 고속 메모리 버퍼 260에 의한 메모리 버퍼 액세스 동작 타이밍을 보여주는 도면이다.
- <142> 상기 도 20을 참조하면, 상기 고속 메모리 버퍼 260에 의한 메모리 버퍼 액세스 동작은 시프트 레지스터 510의 동작에 있어서는 차이가 있지만, 결과적으로 동일한  $\alpha$  메트릭 및  $\beta 1, \beta 2$  메트릭들을 제공함을 알 수 있다.
- <143> 한편 본 발명의 상세한 설명에서는 구체적인 실시 예에 관해 설명하였으나, 본 발명의 범위에서 벗어나지 않는 한도 내에서 여러 가지 변형이 가능함은 물론이다. 그러므로 본 발명의 범위는 설명된 실시 예에 국한되어 정해져서는 안되며 후술하는 특허청구의 범위뿐만 아니라 이 특허청구의 범위와 균등한 것들에 의해 정해져야 한다.

#### 【발명의 효과】

- <144> 상술한 바와 같이 본 발명은 터보 복호기와 동일한 동작주파수의 고속 메모리 버퍼를 사용하여 수신단의 메모리 버퍼부터 터보 복호기로 데이터 비트들을 제공한다. 이러한 본 발명은 수신단의 메모리 버퍼의 동작 주파수를 증가시킬 필요성을 제거하는 것으로, 저전력의 소모가 요구되는 이동통신 환경에 적합한 회로 구현을 가능하게 하는 이점이 있다.

## 【특허청구범위】

## 【청구항 1】

통신시스템의 터보 복호화 장치에 있어서:

데이터 입력을 위한 입력 단자와 데이터 출력을 위한 출력 단자를 구비하고, 입력 데이터 비트들을 상기 입력 단자를 통해 순차적으로 입력 및 시프트시켜 제1 길이의 비트 열들을 형성한 후 상기 제1 길이의 비트 열들을 상기 출력 단자를 통해 순차적으로 출력하는 단방향 시프트 레지스터와,

데이터 입/출력을 위한 제1 단자 및 제2 단자를 구비하고, 상기 입력 데이터 비트들은 상기 제1 길이의 1/2인 제2 길이의 비트들로 이루어지는 그룹들로 분할되고, 상기 분할된 그룹들중 홀수번째 그룹들의 비트들을 상기 제1 단자를 통해 순차적으로 입력 및 시프트시켜 상기 제2 길이의 비트 열들을 형성한 후 상기 제1 단자를 통해 순차적으로 출력하고, 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 순차적으로 입력 및 시프트시켜 상기 제2 길이의 비트 열들을 형성한 후 상기 제2 단자를 통해 순차적으로 출력하는 적어도 하나 이상의 양방향 시프트 레지스터들을 포함하는 메모리 버퍼와;

상기 각 시프트 레지스터들로부터의 출력 비트들을 입력하고 해당하는 델타 메트릭들을 각각 계산하는 제1 메트릭 계산부와,

상기 단방향 시프트 레지스터에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭을 입력하여 알파 메트릭을 계산하는 제2 메트릭 계산부와,

상기 양방향 시프트 레지스터들 각각에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭들을 입력하여 베타 메트릭들을 각각 계산하는 제3 메트릭 계산부와,

상기 알파 메트릭을 입력하고, 또한 상기 베타 메트릭들의 다중화 결과를 입력하여, 상기 각 시프트 레지스터들에 대응하는 로그우도비율(LLR) 값들을 계산하는 제4 메트릭 계산부와,

상기 각 LLR 값으로부터 상기 단방향 시프트 레지스터의 출력을 각각 감산하고, 감산 결과들을 인터리빙/디인터리빙을 위해 출력하는 감산부로 구성되는 시소(SISO) 복호기를 포함함을 특징으로 하는 상기 복호화 장치.

#### 【청구항 2】

제1항에 있어서, 상기 메모리 버퍼는, 상기 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 입력 데이터 비트들을 상기 제1 단자 또는 상기 제2 단자로 인가되도록 제어하기 위한 선택신호들을 상기 양방향 시프트 레지스터들 각각에 제공하는 제어 로직을 더 포함함을 특징으로 하는 상기 복호화 장치.

#### 【청구항 3】

제2항에 있어서, 상기 메모리 버퍼는, 상기 양방향 시프트 레지스터들 각각에 대응하는 역다중화기와, 다중화기를 더 포함하고,

상기 역다중화기는, 상기 입력 데이터 비트들을 입력하기 위한 입력단과, 상기 제1 단자 및 상기 제2 단자에 각각 접속되는 제1 출력단 및 제2 출력단을 구비하고, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 홀수번째 그룹들의 비트들을 상기 제1 출력단을 통해 상기 제1 단자로 인가하고, 상기 짝수번째 그룹들의 비트들을 상기 제2 출력단을 통해 상기 제2 단자로 인가하고,

상기 다중화기는, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제1 단자를 통해 출력되는 비트열들과 상기 제2 단자를 통해 출력되는 비트열들을 다중화하여 상기 제1 메트릭 계산부로 출력하는 것을 특징으로 하는 상기 복호화 장치.

#### 【청구항 4】

제2항에 있어서, 상기 선택신호들 각각은 상기 입력 데이터 비트들이 상기 양방향 시프트 레지스터들 각각에 서로 다른 시점에서 인가되도록 제어하기 위한 신호들임을 특징으로 하는 상기 복호화 장치.

#### 【청구항 5】

제1항에 있어서, 상기 홀수번째 그룹들의 비트들이 상기 제1 단자를 통해 순차적으로 출력됨과 동시에 상기 짝수번째 그룹들의 비트들이 상기 제2 단자를 통해 순차적으로 입력 및 시프트되는 것을 특징으로 하는 상기 복호화 장치.

**【청구항 6】**

제1항에 있어서, 상기 양방향 시프트 레지스터들의 수는 윈도우 수에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 7】**

제1항에 있어서, 상기 제1 길이 및 상기 제2 길이는 윈도우 사이즈와 윈도우 수에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 8】**

제7항에 있어서, 상기 제2 길이는 상기 윈도우 사이즈와 상기 윈도우 수의 곱에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 9】**

제1항에 있어서, 상기 입력 데이터 비트들은 터보 복호기의 클럭 속도로 입력되는 것을 특징으로 하는 상기 복호화 장치.

**【청구항 10】**

통신시스템의 터보 복호화 장치에 있어서:

데이터 입/출력을 위한 제1 단자 및 제2 단자를 구비하고, 입력 데이터 비트들은 미리 설정된 길이의 비트들로 이루어지는 그룹들로 분할되고, 상기 분할된 그룹들중 홀

수번째 그룹들의 비트들을 상기 제1 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제1 단자를 통해 순차적으로 출력하고, 상기 분할된 그룹들중 짝수번째 그룹들의 비트들을 상기 제2 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제2 단자를 통해 순차적으로 출력하는 적어도 하나 이상의 제1 스테이지의 양방향 시프트 레지스터들과,

데이터 입/출력을 위한 제3 단자 및 제4 단자를 구비하고, 상기 제1 단자를 통해 순차적으로 출력되는 비트들을 상기 제3 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제3 단자를 통해 순차적으로 출력하고, 상기 제2 단자를 통해 순차적으로 출력되는 비트들을 상기 제4 단자를 통해 순차적으로 입력 및 시프트시켜 상기 길이의 비트 열들을 형성한 후 상기 제4 단자를 통해 순차적으로 출력하는 제2 스테이지의 양방향 시프트 레지스터를 포함하는 메모리 버퍼와;

상기 각 시프트 레지스터들로부터의 출력 비트들을 입력하고 해당하는 델타 메트릭들을 각각 계산하는 제1 메트릭 계산부와,

상기 단방향 시프트 레지스터에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭을 입력하여 알파 메트릭을 계산하는 제2 메트릭 계산부와,

상기 양방향 시프트 레지스터들 각각에 대응하는 상기 제1 메트릭 계산부로부터의 델타 메트릭들을 입력하여 베타 메트릭들을 각각 계산하는 제3 메트릭 계산부와,

상기 알파 메트릭을 입력하고, 또한 상기 베타 메트릭들의 다중화 결과를 입력하여, 상기 각 시프트 레지스터들에 대응하는 로그우도비율(LLR) 값들을 계산하는 제4 메트릭 계산부와,

상기 각 LLR 값으로부터 상기 단방향 시프트 레지스터의 출력을 각각 감산하고, 감산 결과들을 인터리빙/디인터리빙을 위해 출력하는 감산부로 구성되는 시소(SISO) 복호기를 포함함을 특징으로 하는 상기 복호화 장치.

#### 【청구항 11】

제10항에 있어서, 상기 메모리 버퍼는, 상기 입력 데이터 비트들이 상기 분할된 그룹들중 홀수번째 그룹들의 비트들인지 짝수번째 그룹들의 비트들인지 여부를 판단하고, 그 판단 결과에 따라 상기 입력 데이터 비트들을 상기 제1 단자 또는 상기 제2 단자로 인가되도록 제어하기 위한 선택신호들을 상기 제1 스테이지의 양방향 시프트 레지스터들 각각에 제공하는 제어 로직을 더 포함함을 특징으로 하는 상기 복호화 장치.

#### 【청구항 12】

제11항에 있어서, 상기 메모리 버퍼는, 상기 제1 스테이지의 양방향 시프트 레지스터들 각각에 대응하는 역다중화기와, 다중화기를 더 포함하고,

상기 역다중화기는, 상기 입력 데이터 비트들을 입력하기 위한 입력단과, 상기 제1 단자 및 상기 제2 단자에 각각 접속되는 제1 출력단 및 제2 출력단을 구비하고, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 홀수번째 그룹들의 비트들을 상기 제1 출력단을 통해 상기 제1 단자로 인가하고, 상기 짝수번째 그룹들의 비트들을 상기 제2 출력단을 통해 상기 제2 단자로 인가하고,



상기 다중화기는, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제1 단자를 통해 출력되는 비트들과 상기 제2 단자를 통해 출력되는 비트들을 다중화하여 상기 제1 메트릭 계산부로 출력하는 것을 특징으로 하는 상기 복호화 장치.

**【청구항 13】**

제11항에 있어서, 상기 메모리 버퍼는, 상기 제2 스테이지의 양방향 시프트 레지스터에 대응하고, 상기 제어 로직으로부터 제공되는 해당하는 선택신호에 응답하여 상기 제3 단자를 통해 출력되는 비트들과 상기 제4 단자를 통해 출력되는 비트들을 다중화하여 상기 제1 메트릭 계산부로 출력하는 다중화기를 더 포함함을 특징으로 하는 상기 복호화 장치.

**【청구항 14】**

제11항에 있어서, 상기 선택신호들 각각은 상기 입력 데이터 비트들이 상기 양방향 시프트 레지스터들 각각에 서로 다른 시점에서 인가되도록 제어하기 위한 신호들임을 특징으로 하는 상기 복호화 장치.

**【청구항 15】**

제10항에 있어서, 상기 홀수번째 그룹들의 비트들이 상기 제1 단자를 통해 순차적으로 출력됨과 동시에 상기 짝수번째 그룹들의 비트들이 상기 제2 단자를 통해 순차적으로 입력 및 시프트되는 것을 특징으로 하는 상기 복호화 장치.

**【청구항 16】**

제10항에 있어서, 상기 제1 스테이지의 양방향 시프트 레지스터들의 수는 윈도우 수에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 17】**

제10항에 있어서, 상기 제1 길이 및 상기 제2 길이는 윈도우 사이즈와 윈도우 수에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 18】**

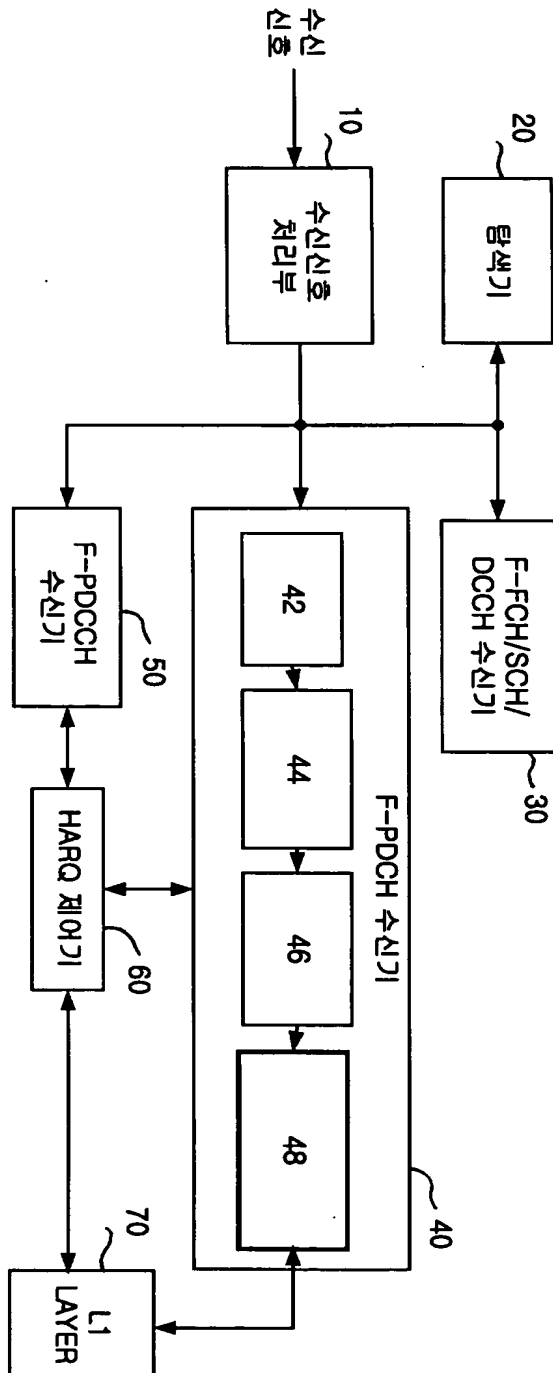
제16항에 있어서, 상기 제2 길이는 상기 윈도우 사이즈와 상기 윈도우 수의 곱에 의해 결정됨을 특징으로 하는 상기 복호화 장치.

**【청구항 19】**

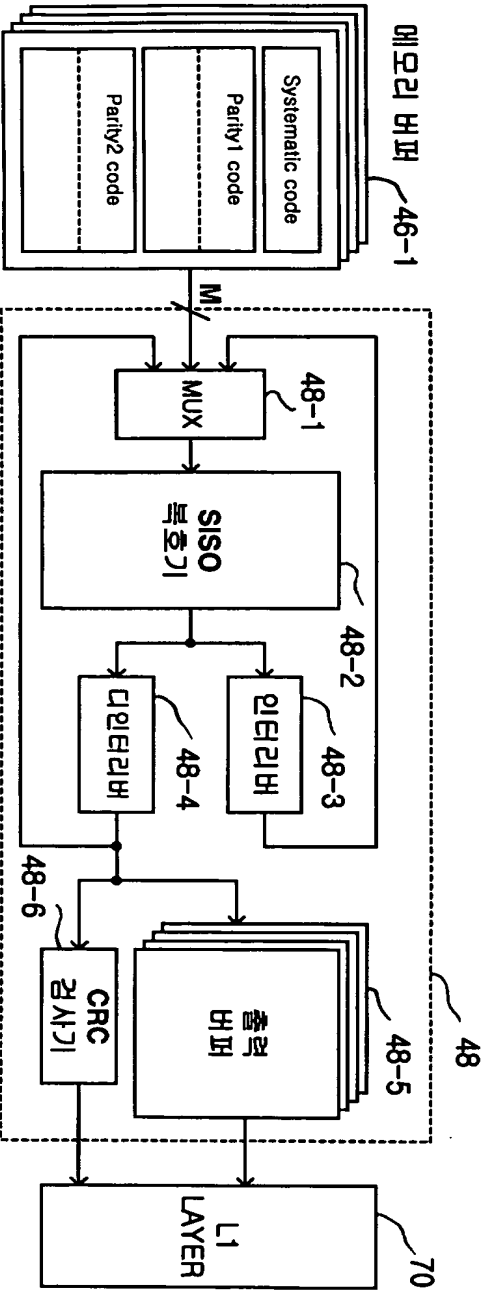
제10항에 있어서, 상기 입력 데이터 비트들은 터보 복호기의 클럭 속도로 입력되는 것을 특징으로 하는 상기 복호화 장치.

【도면】

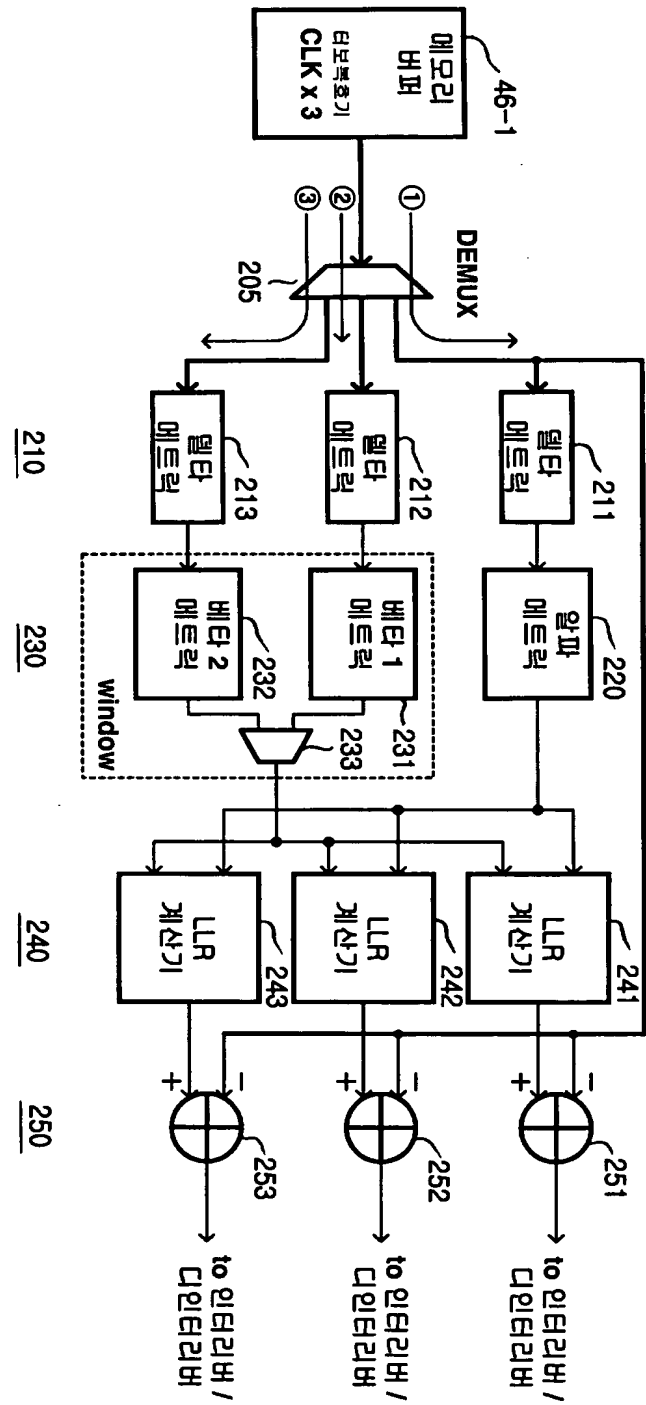
【도 1】



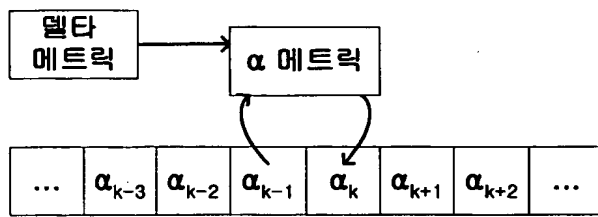
【도 2】



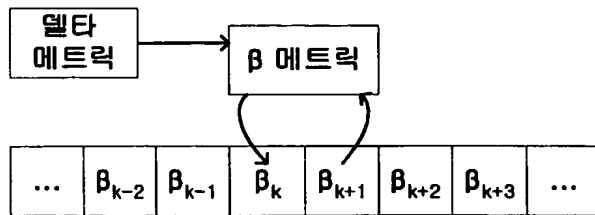
【도 3】



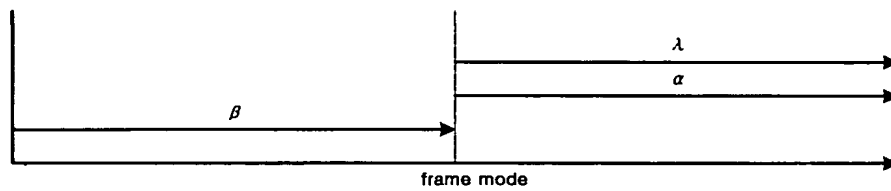
【도 4a】



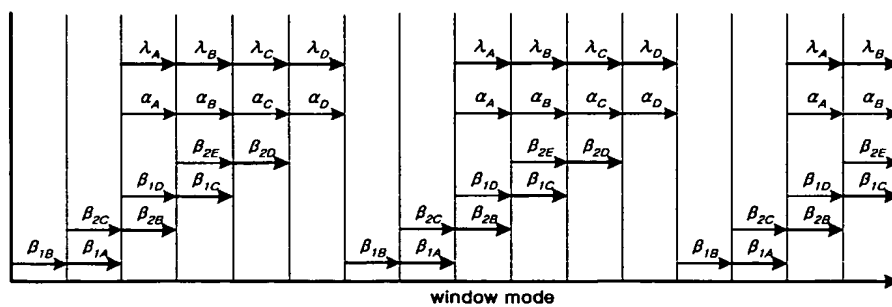
【도 4b】



【도 5a】



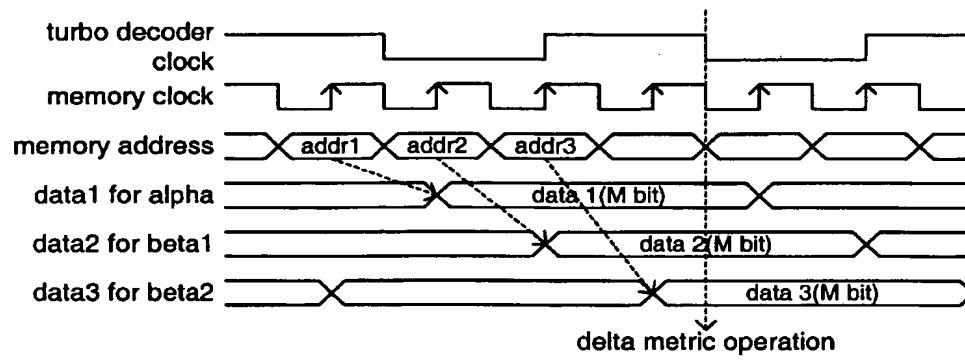
【도 5b】



T1 T2

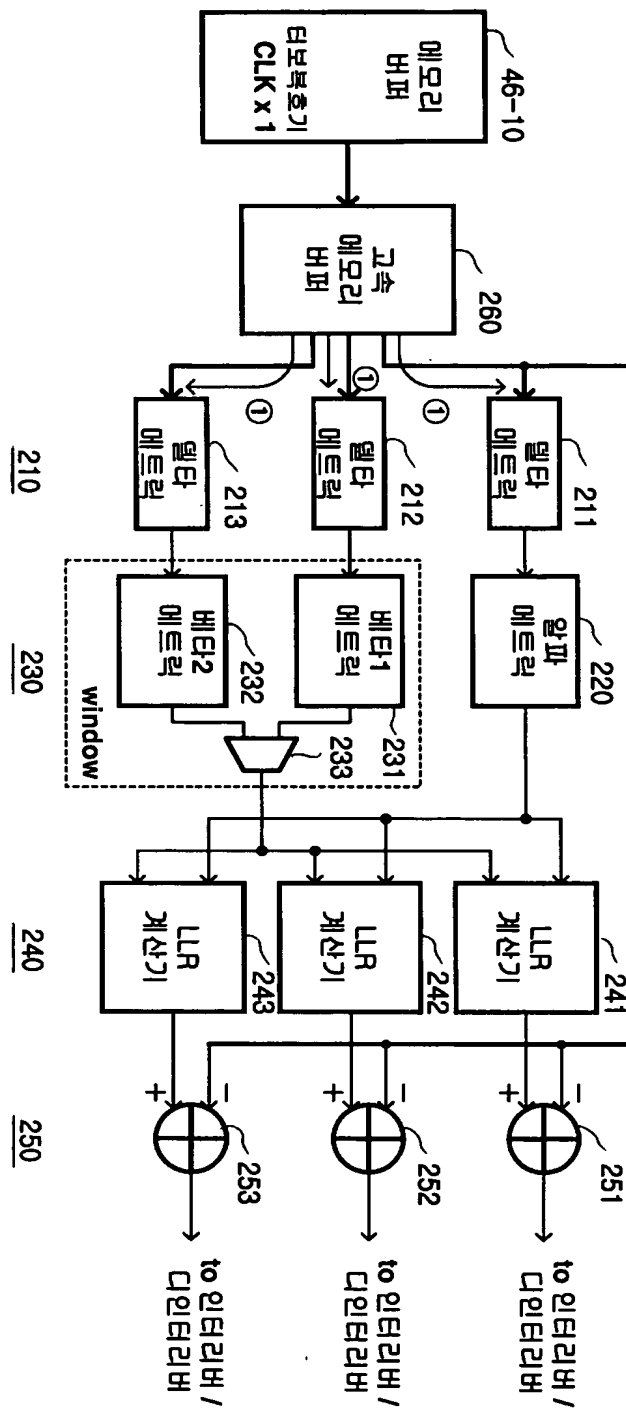
73-61

【도 7】

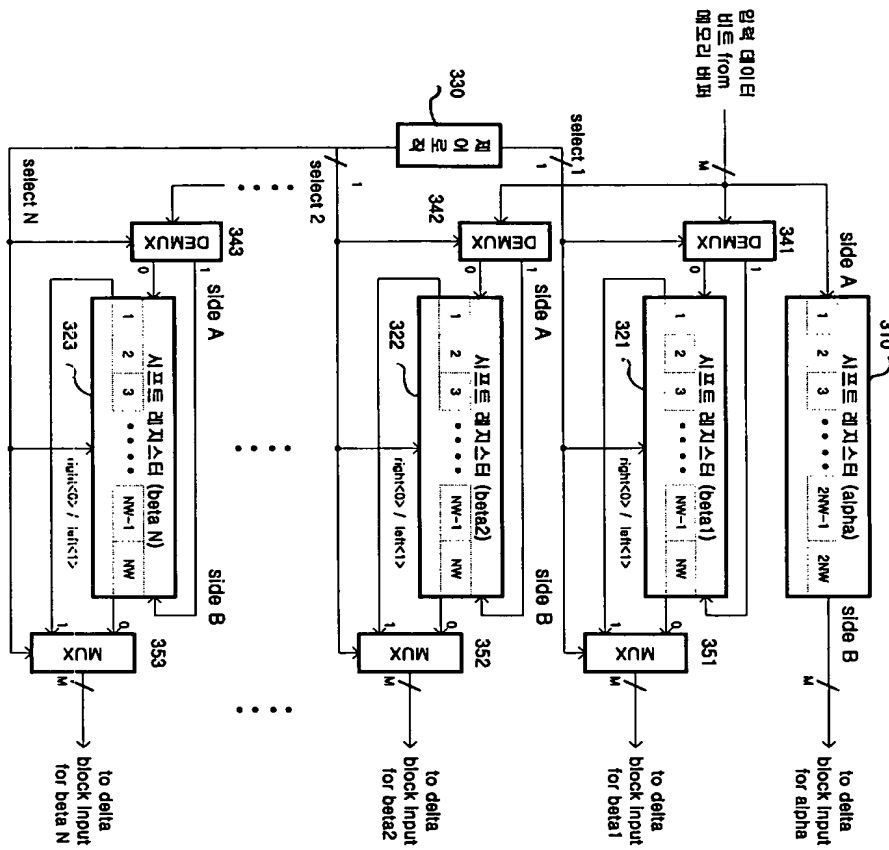




【도 8】

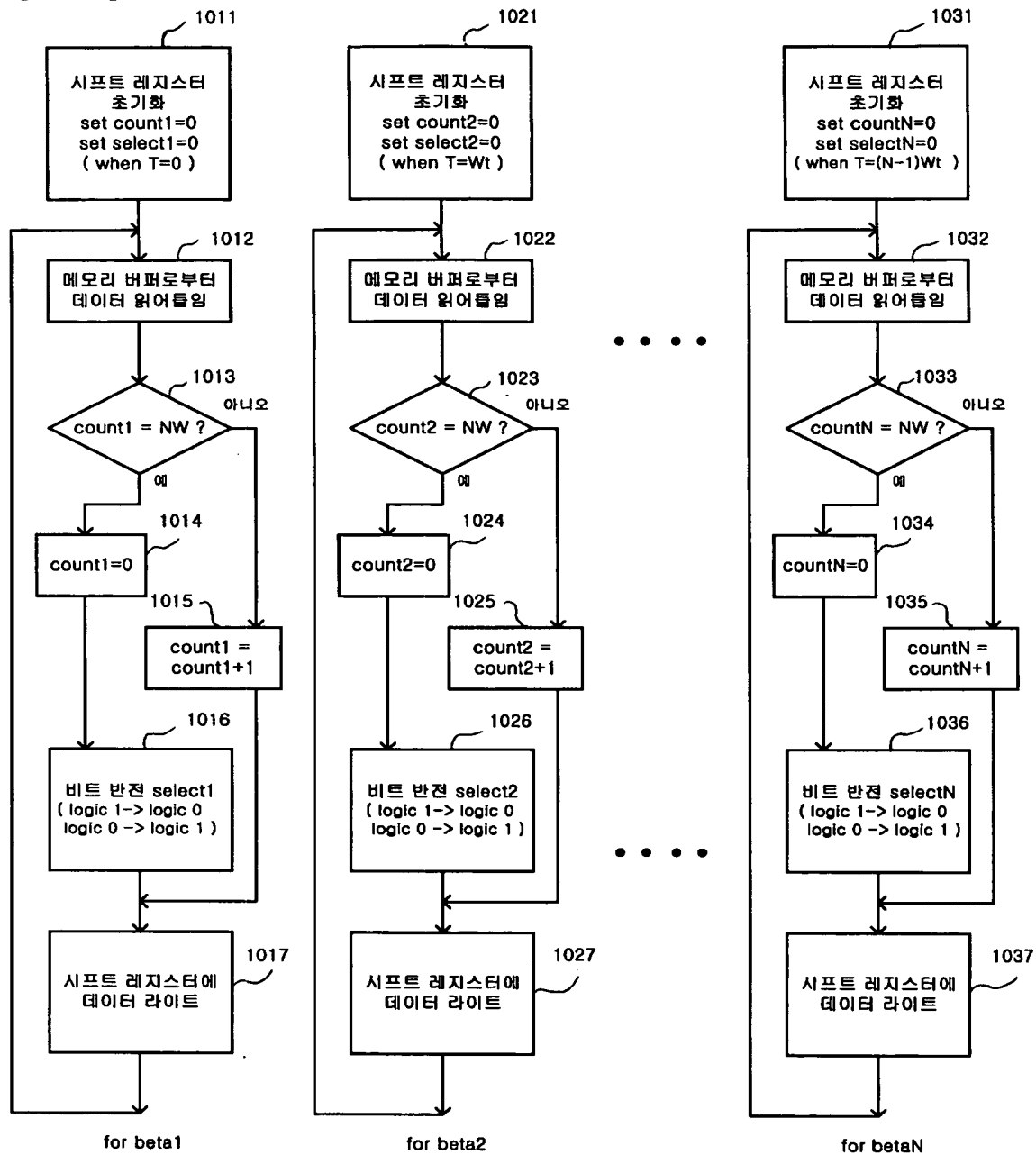


【도 9】



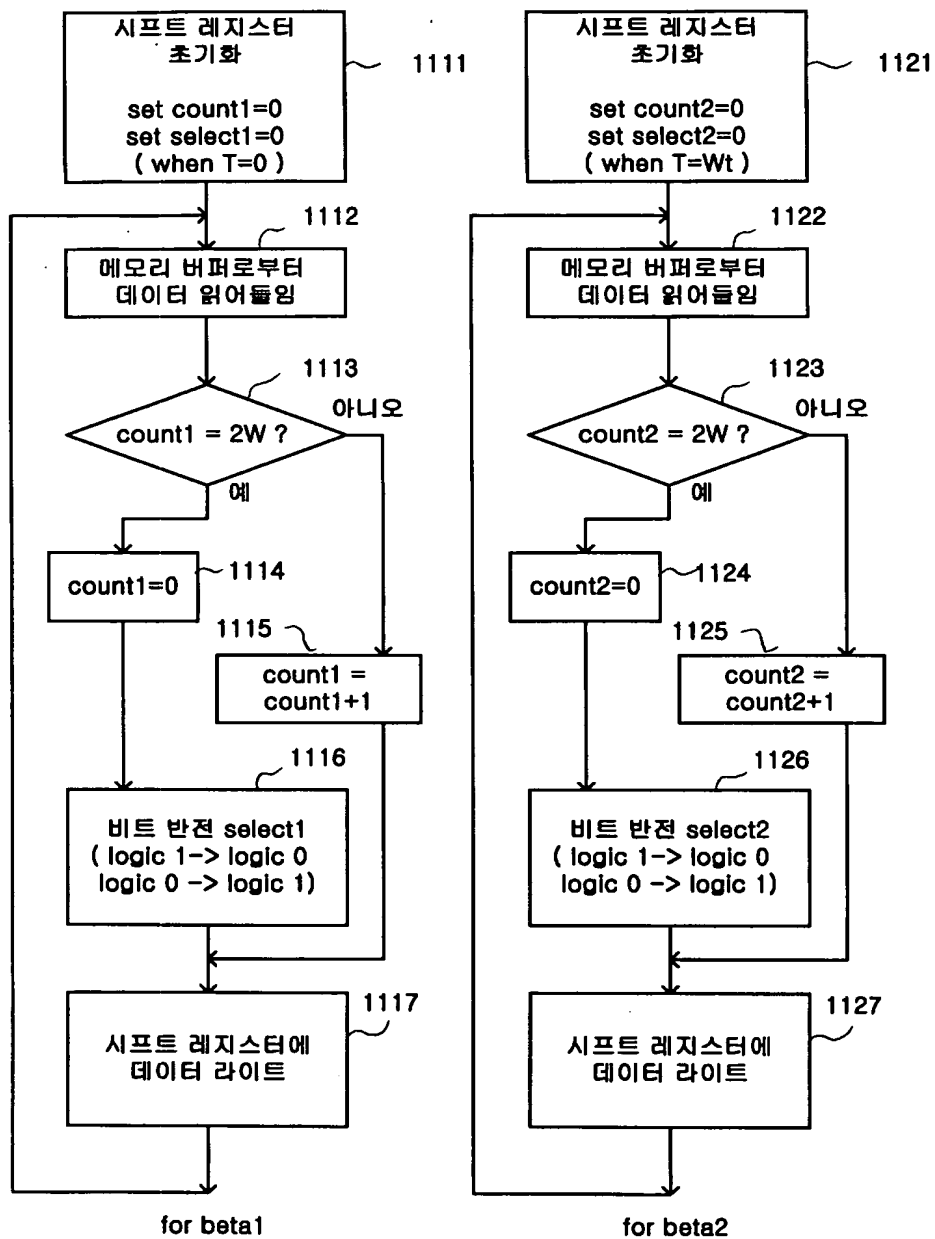
260

【도 10】

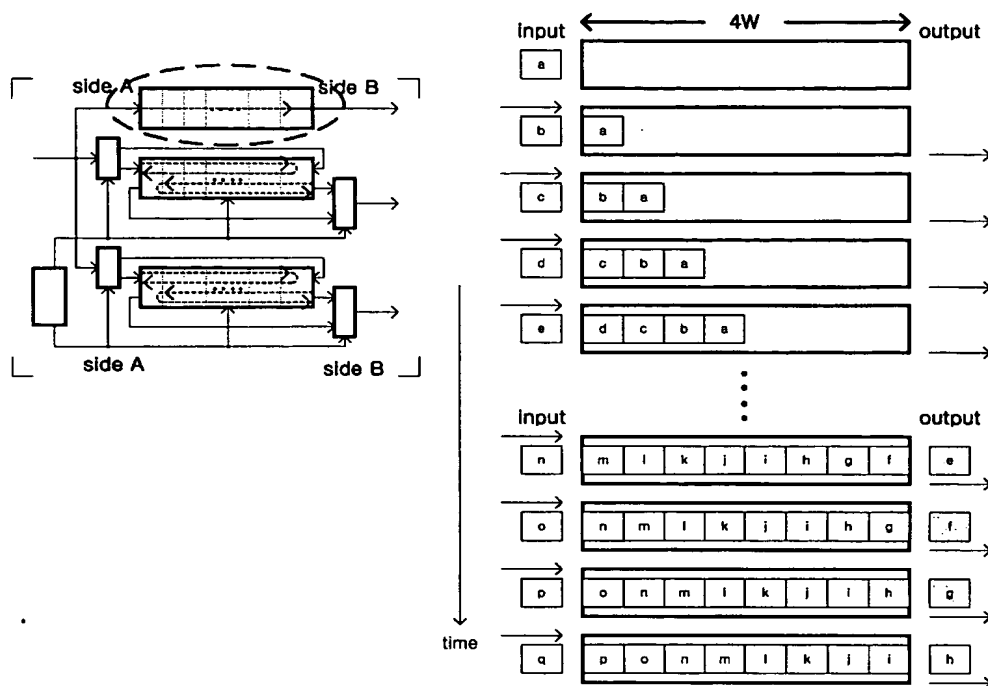




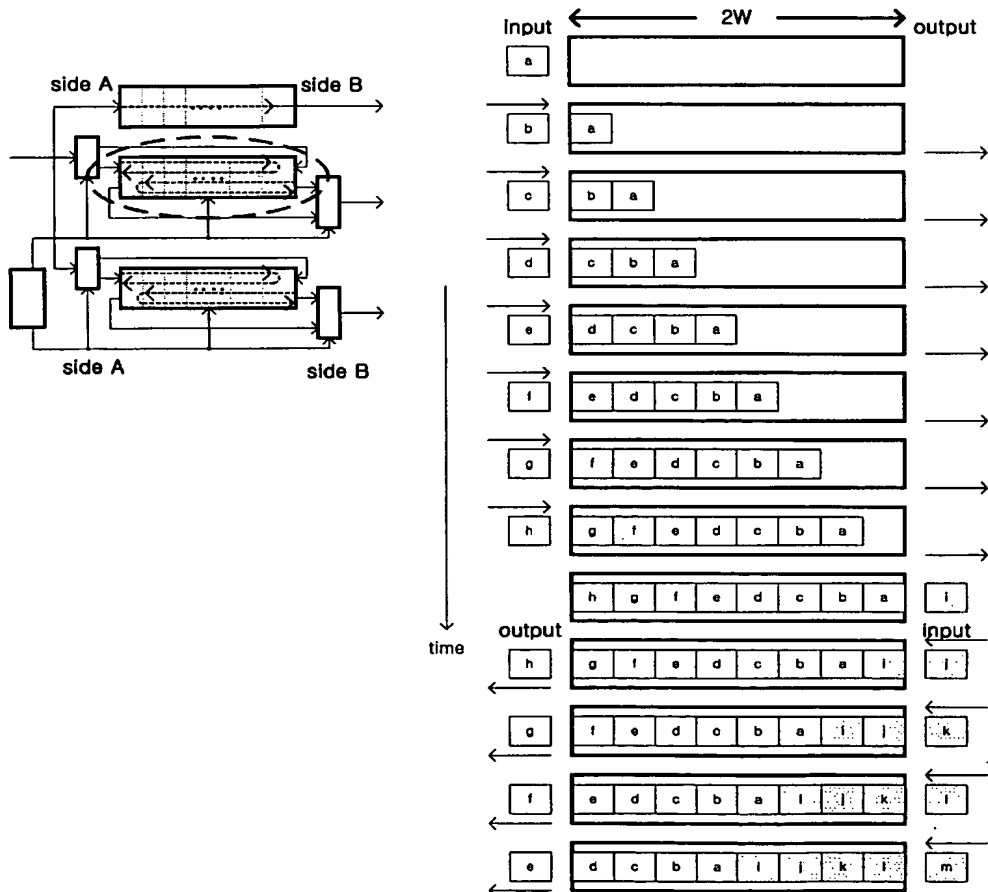
【도 13】



【도 14】



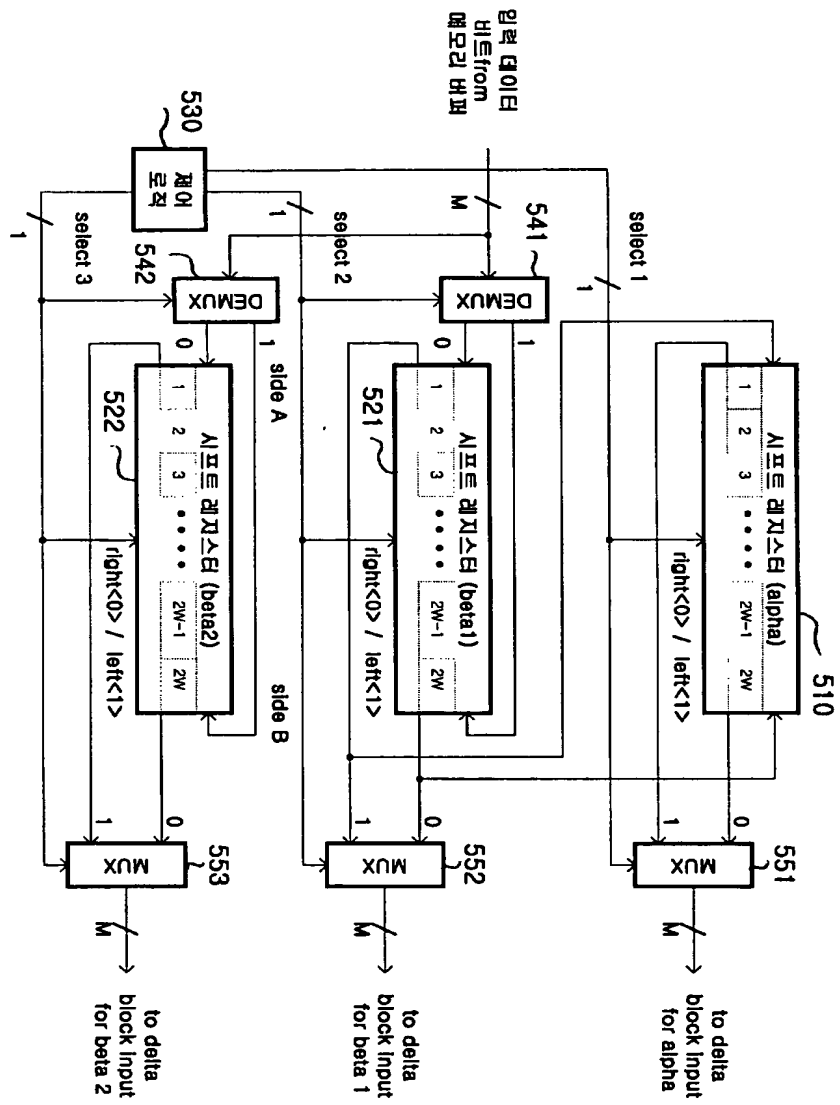
【도 15】



【표 16】

										W = 4		time (transition)																												
QCTC memory buffer-output	side A in	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	at	bt	ct	dt	et	ft	gt						
	side B in	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	at	bt	ct	dt	et	ft	gt						
	input for alpha																																							
	side B out																																							
delta block input for beta1	side A in	a	b	c	d	e	f	g	h																															
	side A out																																							
	side B in																																							
	side B out																																							
delta block input for beta2	side A in					e	f	g	h	i	j	k	l																											
	side A out																																							
	side B in																																							
	side B out																																							
select1 input select2	input clock					logic low						logic high																												
	input clock																																							
	alpha out																	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r					
	beta1 out																																							
beta2 out	input clock																																							
	input clock																																							
	alpha out																																							
	beta2 out																																							

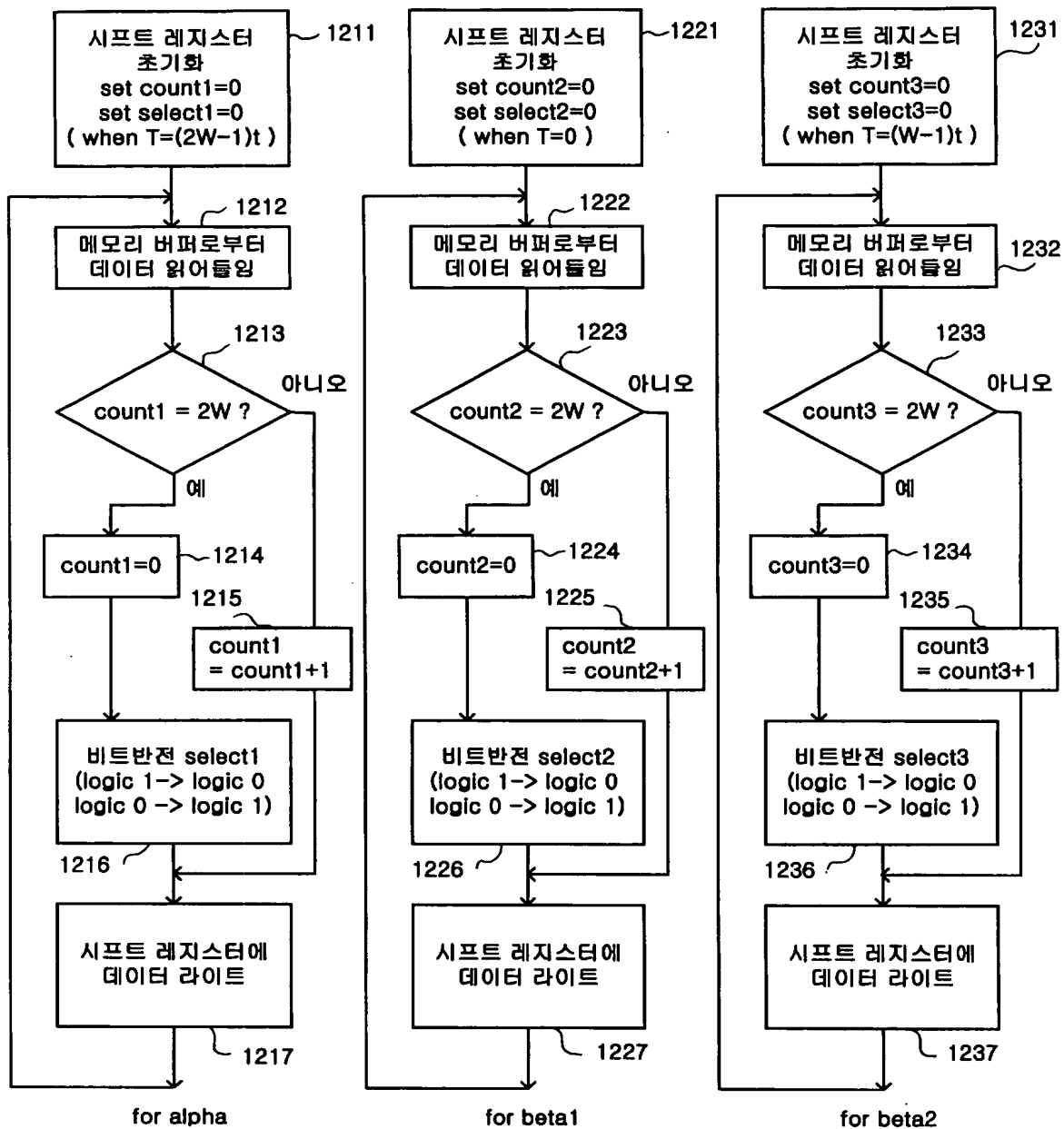
【도 17】



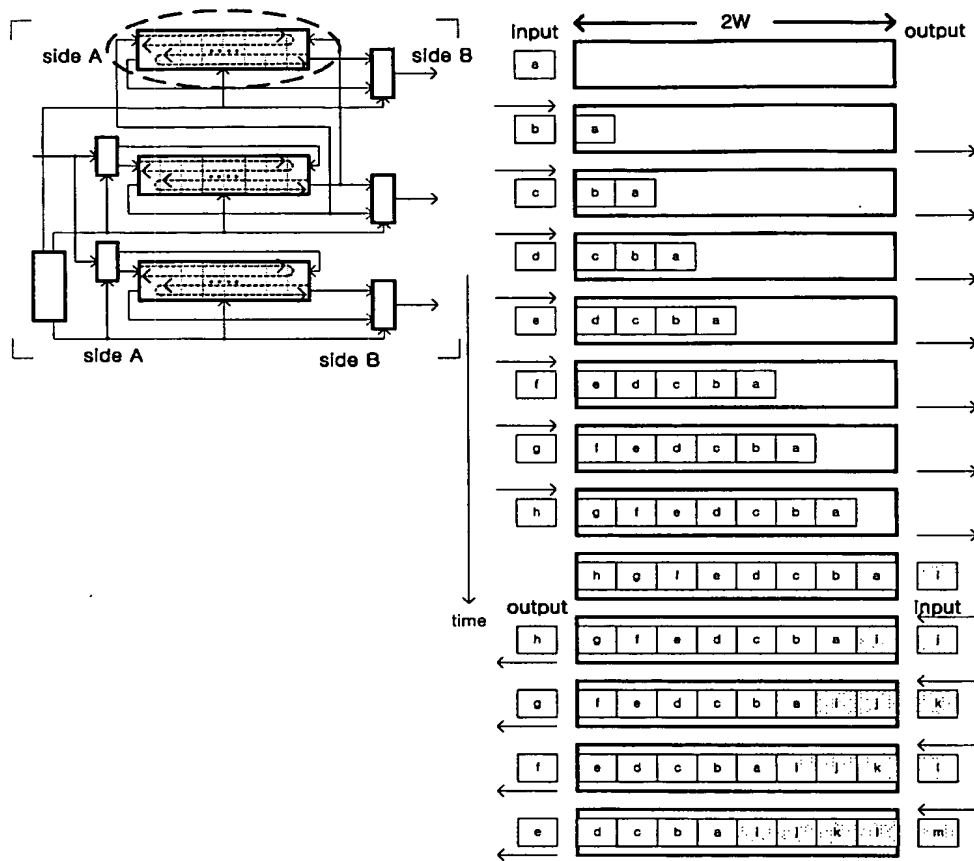
260



【도 18】



【도 19】



		time t (transition)																																	
		W = 4																																	
memory buffer output	side A in	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a1	b1	c1	d1	e1	f1	g1	
	side A in	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a1	b1	c1	d1	e1	f1	g1	
beta block input for alpha	side A in									n	g	t	e	d	c	b	a									x	w	v	u	i	s	r	q		
	side A out																																		
	side B in																	d	o	n	m	i	k	j	i										
	side B out																																		
beta block input for beta1	side A in	a	b	c	d	e	f	g	h									q	r	s	t	u	v	w	x										
	side A out									h	g	f	e	d	c	b	a									x	w	v	u						
	side B in									i	j	k	l	m	n	o	p									y	z	a1	b1	c1	d1	e1	f1	g1	
	side B out																	p	o	n	m														
beta block input for beta2	side A in					e	f	g	h	i	j	k	l									u	v	w	x	y	z	a1	b1						
	side A out									j	k	l	i																	b1	a1	z	y		
	side B in									m	n	o	p	q	r	s	t													c1	d1	e1	f1	g1	
	side B out																					i	s	r	q										
select1	input clock																																		
	input																																		
select2	input clock																																		
	input																																		
select3	input clock																																		
	input																																		
alpha out	metric out																	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
	metric out																																		
beta1 out	metric out																	a	b	c	d														
	metric out																																		